AD-A091 994 MARYLAND UNIV COLLEGE PARK COMPUTER SCIENCE CENTER F/G 9/2
ARCHITECTURE FOR HIGHER LEVEL DIGITAL IMAGE PROCESSING.(U)
FEB 80 A HELLAND, J HUNG DAAG53-76-C-0138

UNCLASSIFIED NL

1 of 2
AD
A091994

ARCHITECTURE FOR HIGHER LEVEL DIGITAL IMAGE PROCESSING

A074851

February 29, 1980

This is the seventh quarterly status report on a program for Image Understanding Using Overlays, conducted by Westinghouse for UMd under Contract DAAG53-76-C-0138 with the U.S. Army Mobility Equipment Research and Development Command, Ft. Belvoir, Virginia 22060.

Prepared for

Computer Science Center
University of Maryland
College Park, Maryland 20742

By

Westinghouse Defense and Electronics Systems Center
Systems Development Division
Baltimore, Maryland 21203

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

(6) ARCHITECTURE FOR HIGHER LEVEL DIGITAL IMAGE PROCESSING.

(12) 142

(11) 29 Feb 80

~~February 29, 1980~~

(9) Quarterly status rept. no. 7,

~~This is the seventh quarterly status report on a~~ program for Image Understanding Using Overlays, conducted by Westinghouse for UMd under Contract (15) DAAG53-76-C-0138 with the U.S. Army Mobility Equipment Research and Development Command, Ft. Belvoir, Virginia 22060.

(10) Arden/Helland Josh/Hung

Prepared for

Computer Science Center
University of Maryland
College Park, Maryland 20742

By

Westinghouse Defense and Electronics Systems Center
Systems Development Division
Baltimore, Maryland 21203

INTRODUCTION

This is the seventh quarterly status report on a program to implement higher level image processing algorithms, being conducted by the Westinghouse Systems Development Division for the Computer Science Center, University of Maryland. Support for the program is provided by the Defense Advanced Research Projects Agency (DARPA) under contract DAAG53-76-C-0138 with the U.S. Army Mobility Equipment Research and Development Command.

The report was prepared by Arden Helland and Josh Hung of Westinghouse. The Westinghouse Program Manager is Dr. Glenn Tisdale. The work was discussed at monthly meetings, held at the University of Maryland (UMd) with Professor Azriel Rosenfeld of UMd and Dr. George Jones of NVEOL.

This report contains results of relaxation processing performed by Westinghouse to demonstrate speed, threshold and convergence properties using test patterns and FLIR imagery. This evaluation was performed on the PDP-VAX GP computer in preparation for the processing of a set of imagery on the Westinghouse Programmable Array Processor (PAP).

ABSTRACT

This report demonstrates results of the one-dimensional, three-label relaxation process ~~as implemented by Westinghouse~~. The process is easily reduced to two labels; this is used to demonstrate the properties of relaxation as analyzed in the sixth quarterly report. The results are shown to confirm the analysis regarding speed of convergence, threshold and stability. Relaxation converged to stable results only when the net alike coefficients are equal; best results are obtained with the fewest iterations when the unlike coefficients are zero. The three label process provided correct segmentation results even when the input image contained less than three distinct subpopulations.

THREE LABEL RELAXATION PROCESS

The gray level relaxation is somewhat revised, compared to the original definition, for several purposes:

1. to improve processor throughput rates
2. to facilitate anticipated evaluation
3. to improve compatibility with the Programmable Array Processor (PAP)
4. to generalize the process to more than two labels

Three labels are used to demonstrate multiple object classes; these correspond to two object polarities and an intermediate level. Typically, these correspond to light and dark objects with intermediate levels representing background clutter, but other interpretations may be useful. The three labels are named bright, clutter, and dark, and are designed B/C/D. Normal Westinghouse convention follows standard TV signal convention as follows: lowest gray values corresponds to lowest video voltage for the brightest areas of the image. Therefore, the highest gray values correspond to dark areas so that dark probability is usually proportional to the gray value, and bright probability is proportional to the negative of gray value. Clutter probability is then computed as the difference between unity and the sum of the bright and dark probabilities.

Input gray values are considered to be fractional values; this is consistent with normal Westinghouse practice for fixed point computers. This means that the user treats the data as if the binary point is to the left of the most significant bit of data. For example, if an eight bit byte contains one gray value, its set of values consists of 0 and positive values from 2^{-8} up to $1 - 2^{-8}$. This also means that partial data is left justified - if there are only six bits of available data in an eight bit byte, they are in the left six bits (next to the implied binary point) with the right two bits unused.

The gray values may be transformed to probabilities for the two active states (bright and dark) by any desired function. A simple function was used for initial testing as indicated in the following table:

G	$0 \leq G \leq .3$	$.3 \leq G \leq .7$	$.7 \leq G \leq 1.0$
FB	$.7 - G$	$.7 - G$	0
FD	0	$-.3 + G$	$-3 + G$

Although these transforms are suitable for direct use as initial probabilities for the two active states, an additional conversion function is used to facilitate adjustment of these transforms to adjust thresholds and to ensure that input probabilities are non-zero. If these transforms are used directly, the resulting threshold can be determined by solving for the gray values at which probabilities for adjacent labels is equal, based on equal net coefficients in the relaxation processing. Assuming that the thresholds are within the $.3 \leq G \leq .7$ range, it can be shown that the residual clutter probability is $1-FB-FD = .4$. Therefore, solving $FB = .4$ results in $G = .3$ as the bright/clutter threshold and $FD = .4$ which results in $G = .7$ as the clutter/dark threshold. The conversion function which allows the thresholds to be adjusted is of the form $P(\alpha) = \alpha min + \alpha IB \cdot F\alpha$ for each active label α(B and D in this case). In general, if the min additive term is not negligible, $\alpha min + \alpha IB$ for each class should not exceed unity for each label; the sum of active probabilities at any gray value should not equal or exceed unity to ensure that the residual clutter probability is positive for all gray values. Initial testing was performed with Bmin and Dmin = .1 and BIB and DIB = .9. This conversion function for initial test results in the following probabilities:

G	$0 \leq G \leq .3$	$.3 \leq G \leq .7$	$.7 \leq G \leq 1.0$
PB	$.73 - .9G$	$.73 - .9G$	.10
PD	.10	$-.17 + .9G$	$-.17 + .9G$
PC	$.17 + .9G$	.44	$1.07 - .9G$

These revised values can be solved for thresholds by setting clutter probability equal to bright and dark probabilities; the results are bright threshold gray value = .32 and dark threshold gray value = .68. Of course, if the desired gray value thresholds are known, they may be used to solve for the required αmin and αIB values. Generally, for probabilities positively proportioned to gray values, the threshold is increased by increasing αIB, and is reduced by increasing αmin (correspondingly reducing αIB to maintain $\alpha min + \alpha IB \leq 1$).

The image size is determined by the input data format. Because the relaxation process uses a 3x3 neighborhood, the number of pixels processed is two less than the image size in each direction. Therefore, means must be provided to preserve image size throughout the relaxation processing. This is accomplished by a process called "border unfolding". Every pixel on the external border cannot be the center pixel of a 3x3 neighborhood, so it is unaffected by relaxation iterations. Therefore, after each iteration, the data for each pixel that was on the border of the region processed is moved out to its neighboring pixel on the image border. For example, each pixel in row 1 is set equal to the relaxation results for each corresponding pixel from row 2. Likewise, the last processed row is unfolded to the last image row. Then, the first and last columns are unfolded in the same manner. It may be noted that each pixel on the corner of the region processed by relaxation will be unfolded in both directions; therefore, the unfolding will result in all four pixels in each corner being equal. The general form of the relaxation processing is structured to consist primarily of multiplication and addition; this is done in anticipation that this will simplify operation on the array processor. Division is avoided to the maximum extent possible since the current array processor does not include the capability to perform division in the Vector Array Processor (VAP); division must be performed by table look-up in memory or by the Control Arithmetic Processor (CAP).

The relaxation processing structure was also modified to provide consideration for multiple adjacent labels for the more than two label case. Since the gray level data is one-dimensional, there are two "external" labels (bright and dark in this case); these have maximum probabilities at the upper and lower limits of gray values, respectively. However, for this case of three labels, there is an "internal" label called clutter. This label is adjacent to both the bright and dark labels with respect to its position along the gray level dimension. This relationship is considered in the structure of the relaxation processing by adding a term in the updated clutter probability that is proportional to the product of the two adjacent label probabilities. This function is called joint bright, dark enhancement of clutter; its primary purpose was intended to reduce ambiguity between non-adjacent labels, particularly when joint relaxation with edge enhancement is used with gray level relaxation.

The intermediate summation function for the bright label is as follows:

SBi=KB*Pib*(Tbj + N*Kbc)

KB is the net bright coefficient, rbb-rbc

Kbc is the normalized bright/clutter coefficient, rbc/KB=rbc/(rbb-rbc)

Pib is the bright probability of the ith center pixel

Tbj is the sum of the bright probabilities for all j neighboring pixels

N is the number of neighboring pixels (N=8 for current processing)

The intermediate summation function for the dark label is the same as for bright, except the terms are dark probabilities and coefficients. It may be noted that the speed of convergence to the bright label was defined as Cb=(rbb-rbc)/rbb; it can be shown that this is equivalent to Cb=1/(1+Kbc). Likewise, Cd=1/(1+Kcd). Normal AUTO-R processing is performed with Kb=Kd=1 and Kbc and Kcd=0 for maximum speed, which gives greatest change per iteration and final, stable results in the fewest iterations.

The intermediate summation function for the clutter label is of the same form as for bright or dark, with the addition of the joint bright, dark enhancement term as follows:

Sci=Kc*Pic*(Tcj + N*Kbcd + DELTA*Sbi*Sdi)

Kc is the net clutter coefficient, rcc-rbc-rcd

Kbcd is the normalized sum of adjacent label coefficients, (rbc+rcd)/Kc

Pic is the clutter probability of the ith center pixel

Tcj is the sum of the clutter probabilities for all j neighboring pixels

N is the number of neighboring pixels

DELTA is the joint bright, dark enhancement of clutter coefficient; an initial value of 0.5 was used.

The updated probability is defined as the ratio of the appropriate intermediate summation function to the total of the summation functions. This is implemented by computing the inverse of the total once so that the updated probability for each label is computed by multiplication.

Although this relaxation process is structured for three labels, it is easily adapted for two label processing by setting the αmin and αIB adjustment terms to zero for the class to be deleted. Similarly, the polarity of the bright and dark labels may be reversed by interchanging which probability is positively proportional to gray values.

OCTAGON TEST PATTERN AND THE TWO LABEL RELAXATION PROCESS

A test pattern was developed to demonstrate the convergence properties of relaxation processing. The gray levels of the octagon are shown in Table 1, which has the following characteristics: (a) The gray level of the pixels in the central portion of the octagon starts at a high of 0.9 and tapers off to a low of 0.1. (b) The gray level of the pixels in both the left top and the left bottom corners also starts at 0.9 and gradually tapers off to 0.1. (c) In the right top corner, the gray level starts at 0.6 and tapers off to 0.1. (d) In the right bottom corner, the gray level starts as relatively low at 0.3 and tapers off to 0.1. The test areas in the corners are intended to demonstrate threshold response and stability of the border unfolding.

The central region in the octagon may be used to demonstrate relaxation response to a linear boundary. By using a set of compatibility coefficients to give a stable threshold, the relaxation updating process will drive the bright probabilities, for those pixels whose neighbors' bright probabilities are originally higher than the threshold to unity. At the same time, the process will drive the bright probabilities, for those pixels whose neighbors' bright probabilities are lower than the threshold, to zero. The process is stable with respect to the pixels on the linear boundary which separate the two contrasting labels.

Three sets of compatibility coefficients (shown in Table 2) were used for the two label (bright/clutter) relaxation process to demonstrate different effects on the thresholds, and the convergence speeds. In Table 2, cases (A) and (B) have two different sets of compatibility coefficients which yield a stable threshold (0.5) but two different convergence rates. Case (A) has faster convergence speed with respect to case (B). Case (C) has a set of compatibility coefficients yielding an unstable threshold (0.16 bright threshold); these compaitiblity coefficients are taken from the average of the coefficients used in TR795 (Danker).

TABLE 1. GRAY VALUES OF THE OCTAGON TEST PATTERN

.90	.90	.80	.60	.40	.20	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.20	.30	.40	.50	.60
.90	.90	.80	.60	.40	.20	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.20	.30	.40	.50	.60
.80	.80	.60	.40	.20	.10	.10	.10	.10	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.10	.10	.10	.10	.20	.30	.40	.50
.60	.60	.40	.20	.10	.10	.10	.10	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.10	.10	.10	.10	.20	.30	.40
.40	.40	.20	.10	.10	.10	.10	.10	.10	.10	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.20	.20	.20	.10	.10	.10	.10	.20	.30
.20	.20	.10	.10	.10	.10	.20	.20	.20	.30	.30	.40	.40	.40	.40	.40	.40	.40	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10	.10	.20
.10	.10	.10	.10	.10	.20	.20	.20	.30	.30	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10	.10
.10	.10	.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10
.10	.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10	.10
.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.60	.60	.60	.60	.60	.60	.60	.60	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10
.10	.10	.20	.20	.30	.30	.40	.40	.50	.50	.50	.60	.60	.60	.60	.60	.60	.60	.60	.60	.60	.50	.50	.50	.40	.40	.30	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.50	.60	.60	.70	.70	.70	.70	.70	.70	.70	.70	.60	.60	.50	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.70	.80	.80	.80	.80	.80	.70	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.80	.80	.80	.80	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.90	.90	.90	.90	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.90	.90	.90	.90	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.90	.90	.90	.90	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.90	.90	.90	.90	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.80	.80	.80	.80	.80	.80	.80	.80	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.60	.60	.70	.70	.80	.80	.80	.80	.80	.70	.70	.60	.60	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.40	.40	.50	.50	.50	.60	.60	.70	.70	.70	.70	.70	.70	.70	.70	.60	.60	.50	.50	.50	.40	.40	.30	.20	.20	.10
.10	.10	.20	.20	.30	.30	.40	.40	.50	.50	.50	.60	.60	.60	.60	.60	.60	.60	.60	.60	.60	.50	.50	.50	.40	.40	.30	.30	.20	.20	.10
.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.60	.60	.60	.60	.60	.60	.60	.60	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10
.10	.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10	.10
.30	.30	.10	.10	.20	.20	.20	.30	.30	.40	.40	.50	.50	.50	.50	.50	.50	.50	.50	.50	.50	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10
.50	.50	.30	.10	.10	.20	.20	.20	.30	.30	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10	.10
.70	.70	.50	.30	.10	.10	.20	.20	.20	.30	.30	.40	.40	.40	.40	.40	.40	.40	.40	.40	.30	.30	.20	.20	.20	.10	.10	.10	.10	.10
.90	.90	.70	.50	.30	.10	.10	.10	.10	.10	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.30	.20	.20	.20	.10	.10	.10	.10	.10	.20
.90	.90	.70	.50	.30	.10	.10	.10	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.10	.10	.10	.10	.10	.20	.20
.90	.90	.90	.90	.70	.50	.30	.10	.10	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.20	.10	.10	.10	.10	.10	.20	.20	.30
.90	.90	.90	.90	.90	.70	.50	.30	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.20	.20	.30	.30
.90	.90	.90	.90	.90	.70	.50	.30	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.10	.20	.20	.30	.30	.30

Table 2. Three different compatibility coefficient sets yield different thresholds (Tb) and convergence speecs.

	Case (A)	Case (B)	Case (C)
rbb	1.0	1.0	1.90
rbc	0.0	0.9	0.83
rcc	1.0	1.0	1.03
Tb	0.5	0.5	.16
Cb	1.0	0.1	.46
Cc	1.0	0.1	.19

RESULTS FOR COMPARISON OF SPEED CONVERGENCE

A compilation of several continuous samples crossing a linear boundary in the octagon was made for cases (A) and (B) of Table 2. The probabilities of these samples for several iterations are listed in Tables 3A and 3B. The minimum value allowed is .01 to avoid multiplication by zero. The pixels in row 11 of both Table 3A and 3B may be compared to indicate an obvious difference of convergence speed to the bright label. Likewise, row 6 shows the comparison for convergence to dark; rows 6 and 11 add to unity (within roundoff error) which indicates that convergence speeds to the opposite labels are equal. It is also obvious that rows 8 and 9 define the linear boundary separating the two contrasting regions in both tables. Rows 8 and 9 were initially at threshold levels and remain as a stable boundary definition throughout the relaxation iterations.

Table 3

(A) Convergence Speed Cb=1.0

	Pib (Pixel Coordinate)					
Iteration	Pib(15,6)	Pib(15,7)	Pib(15,8)	Pib(15,9)	Pib(15,10)	Pib(15,11)
0	.40	.40	.50	.50	.60	.60
1	.27	.34	.46	.54	.66	.73
2	.11	.23	.41	.59	.77	.89
3	.02	.09	.32	.68	.91	.98
13	.01	.01	.01	.98	.99	.99
103	.01	.01	.02	.99	.99	.99

(B) Convergence Speed Cb=0.1

	Pib (Pixel Coordinate)					
Iteration	Pib(15,6)	Pib(15,7)	Pib(15,8)	Pib(15,9)	Pib(15,10)	Pib(15,11)
0	.40	.40	.50	.50	.60	.60
1	.39	.40	.50	.50	.60	.61
2	.39	.39	.50	.50	.61	.61
3	.38	.39	.49	.51	.61	.62
13	.29	.34	.47	.53	.66	.71
103	.02	.01	.07	.90	.99	.98

Figure 1 (b-f) and Figure 2 (b-f) show iterations 1, 2, 3, 13, and 103 of the two label (bright/clutter) relaxation process using the compatibility coefficients of cases (A) and (B) in Table 1. The selection rules for plotting were as follows:

1) double "BB" characters for any pixel with probability greater than 0.5 being bright,
2) a single "B" character for a pixel with $0.3 < Pib < 0.5$
3) no character (blank) for $Pib < 0.3$
4) "CC", "C" or blank for the same limits for clutter probability

The initial classifications of the test octagon for the relaxation process is shown in both Figure 1 and Figure 2. A solid section with double "BB" characters exists in the center of the octagon and an undecided region with

single "B" characters surrounds the solid section.

The boundary (undecided) region for case (A) (Cb = 1.0) is reduced by the first iteration; the same area for case (B) (Cb = 0.1) appears unchanged at the end of first iteration. The undecided region for case (A) has completely disappeared by the third iteration, while the same area for case (B) appears unchanged. Figure 2 shows eventual definition of the boundary region for case (B) by the end of iteration 103.

Figure 3 (a-f) shows iterations 1, 2, 3, 13, and 103 of the two label cases (bright/clutter) relaxation process using the compatibility coefficients set with the unstable threshold of 0.16 - case (C) of Table (1). The results show that the probability of the original octagon and the four corners grows gradually for every iteration. The regions with double "BB" characters keep expanding. At the end of iteration 103 the probability of the whole frame has converged to the upper limit of 0.99. This result is not surprising because the compatibility coefficient rbb is 1.93, which is greater than rcc which is 1.03. This difference increases the bright probability in a boundary region to unbalance the probabilities of the boundary, moving it into its neighboring regions.

From the results of the above three figures, we may draw the following conclusions. By choosing stable threshold coefficients, the relaxation process segments (enhances) regions (at least if they have linear boundaries). The process drives the probabilities of regions to certain labels. The boundary between regions is stable relative to the original shape. However, if coefficients yielding an unstable threshold are chosen, the relaxation process will only enhance the image at the initial states. After several iterations, regions are either enlarged or diminished from their original size. The unstable relaxation process eventually drives the probabilities of the entire frame toward one label. The two cases of a stable threshold, but different speed of convergence showed that the faster set of stable coefficients produced equivalent results with fewer iterations than the other set for the same relaxation process; both converged to the same stable results.

A REAL IMAGE FRAME AND THE RELAXATION PROCESS

The results of the previous section show the advantage of the stable

threshold coefficients over the unstable ones, and the difference between the fast and slow convergence coefficient sets. A frame of real imagery (the same as used in TR795) is chosen to evaluate with the same coefficient sets which are given in Table 2 to compare results with these obtained for the octagon test case.

The results of Figure 4 and Figure 5 show the difference between the fast and slow convergence speed of the two stable coefficients. A blob and background noise is shown at the zero iteration (see Figures 4 and 5). The shape of the blob for both cases diminishes very little from the original shape even at the end of 100th iteration. However, the background noise for both cases has completely disappeared. Two important facts are observed: (1) the shape of the blob in case (A) at the 8th iteration is about the same as the blob in case (B) at the 100th iteration; (2) the background noise for case (A) has disappeared except at the left bottom corner while the background noise of case (B) persists through iteration 8. This again demonstrates that the faster coefficients obtain equivalent results in fewer iterations.

Figure 6 shows the result of the relaxation process with the unstable coefficients. The original blob is small and somewhat uncertain, but grows to a larger size at the 8th iteration. At the 100th iteration the bright probabilities of the blob have unbalanced the probabilities of its neighbors and the blob's boundary becomes unidentifiable. All regions with initial bright probability above the threshold have converged to the bright label and the bright regions continue to expand. The threshold of .16 resulted from deriving compatibility coefficients from the same image, which had very few bright pixels. This threshold is so much lower than that necessary to segment the target blob that many non-target background regions are also segmented to the same label as the target. The first eight iterations shown in Figure 6 correspond to the Figure 6 case shown in TR795. Comparison shows that the results are quite similar, considering that there are differences in scaling, partly due to the limitation of only three states in the clutter printout ("CC", "C" and "blank" divided at bright probabilities of 0.3 and 0.5). The tendency for the bright regions to increase without limit is evident by the 8th iteration and well confirmed by iteration 100. These results indicate that "best" segmentation results of unstable relaxation are only temporary. Inspection

of Figure 6 indicates that "best" results occur at about iteration 4; further processing causes continued loss of blob definition. In contrast, the fast and stable relaxation processing of Figure 4 achieved results comparable to the "best" of Figure 6 at the first iteration and virtually complete segmentation of the target blob of iteration 2. It may be noted that the unstable relaxation never achieved the complete segmentation of just the target blob; stable relaxation converged to complete segmentation with the only effect of further iterations being that the corners and edges are somewhat smoothed.

Figures 7, 8, and 9 show the effects of increasing the bright threshold by reducing the bright initial bias (BiB) multiplicative factor. The successive bias values are .85, .65, and .50, corresponding to relaxation thresholds of .19, .25, and .32, respectively. The compatibility coefficients remain the same as for Figure 4, so the process remains slow and unstable. The effect of the lower bias factors is evident by the successively smaller initial region labeled bright and less of the background region initially labeled bright. The highest bright relaxation threshold (.32, which results from the .50 bias factor) gives final results which are closest to the relaxation threshold of .50 which resulted from the stable coefficients as shown in Figure 4. Of course, the unstable process results in continued growth of the bright regions in Figures 6 through 9, so it is difficult to directly compare results. Comparative results of relaxation iterations shows that the segmentation between the target blob and background is best with the highest threshold in Figure 9. However, the bias factor is so low that the target blob is tiny and virtually indistinguishable at the early iterations; the blob must "grow" to its "proper" size and shape. Reasonable blob size and shape is finally obtained by iteration 8 in Figure 9; roughly comparable results are obtained somewhere between iterations 4 and 8 for the other figures, but with more of the background labeled bright. None of the results in Figures 6 through 9 appear to be as "good" as that achieved by iteration 1 or 2 (and the following stable iterations) in Figure 4.

It is also of interest to compare the results of Figure 9 with the "borderness" processing using joint edge/no edge, light/dark relaxation shown in Figure 12 of TR795. Although joint relaxation appeared to improve the segmentation of the target blob, growth into the background and emergence of points with bright labels in the background is not significantly reduced,

based on the results of Figure 10 in TR795. A modification called "borderness" was tried, which improved initial results. However, "borderness" included the use of an initial bias factor to the entire image before adding the borderness values to edges. The borderness bias value used in Figure 12 of TR795 was 0.5, so it is of interest to compare this to Figure 9 of this report which also used a bias value of 0.50. Comparison of these two figures shows virtually identical results at each iteration, within the ability of the three printed characters to represent gray scale. It appears reasonable to deduce from the similarity of these results that the use of the "borderness" concept, as well as joint relaxation, has no significant effect on the results. Conversely, adjustment of the relaxation threshold provides the major known change in segmentation results at intermediate iterations. Of course, it has been previously shown that only stable coefficients (Kb=Kc in this case) allow relaxation to converge to useful results. The continued growth even with the bias value of 0.50 is shown by iteration 100 in which the target blob has expanded to a large ellipsoid. Iteration 100 also indicates that there remained one background region in the lower left corner that was above the relaxation threshold. The unstable coefficients cause this region to expand continuously in Figure 9. This same region was evident in Figure 4 (especially note iteration 2), but the stable process is able to reject it rapidly (due primarily to lack of surrounding regions with alike levels), so that the probability has dropped below the "blank" character level of 0.3 by the fourth iteration.

It may be noted that alghough the different bias values shift the relaxation threshold with respect to the input gray values, the <u>probability</u> threshold remains defined by the compatibility coefficients, which is, in terms of the K factors, $Tb=1/(1+Kb/Kc)=1/(1+1.07/0.70)=1/6.35=.1575$. Therefore, gray values between approximately .32 and .60 will be transferred to probabilities between .16 and .30. These values are above the probability threshold, but below the single "B" character threshold, so they will remain as "blank" characters until the relaxation iterations increase the probabilities above .3 for single "B" and .5 for double "BB" characters. This condition is apparently what occurs in the lower left corner for Figure 9. In comparison, Figure 4 has a probability threshold of 0.50 for stability and unity bias so that the gray level threshold is also 0.50. Since this is

also the threshold for double "BB" characters printout, all "BB" characters in the initial printout represent pixels with initial gray values above the relaxation threshold. Except for the target blob region, the only regions above threshold are isolated pixels in the lower left corner at the border of the image.

The following conclusions are based on observation of the results of Figure 4 through 9:

1. The image frame contains a relatively high contrast target blob and a moderately noisy background.
2. The image was processed by stable relaxation at an equivalent gray level threshold of 0.50 with excellent segmentation by iteration 2 and no significant deterioration demonstrated through iteration 100.
3. The image was processed by the original unstable relaxation process at an equivalent gray level threshold of 0.16 and 0.32 which corresponds to the results in Figures 6 and 12 of TR795 for light/dark relaxation and joint edge/no edge, light/dark relaxation using initial borderness. The results were nearly identical to those in TR795; the low threshold results in segmentation of background regions, the higher threshold takes much longer to obtain a "reasonable" segmentation of the target blob.
4. The image was also processed by the original unstable relaxation process at equivalent intermediate thresholds of 0.19 and 0.25 with results intermediate between those described above. The results of all of the unstable processing deteriorates by blob growth for later iterations; target shape is virtually undistinguishable at iteration 100 for all cases. The original process gives generally the "worst" results; virtually all of the image frame is segmented as target by iteration 100.

RESULTS OF THREE LABEL RELAXATION

Having presented results to demonstrate effects of stable and non-stable thresholds, fast and slow convergence speed coefficients for the two label

relaxation process, we shall show results of three label relaxation applied to the previous images plus a new test image called "MODPOT 26". Since the previous tests indicated that unity coefficients (the AUTO-R process) provided clearly superior results, three label relaxation was performed using only the unity coefficients. No attempt was made to adjust thresholds or to normalize the input data. The initial probability classifications for these three cases are defined by the Westinghouse convention as described in the first section of this report. The selection rules for plotting are the same as in the preceding sections.

A. Octagon Test Pattern, Figure 10

Iteration 0: Figure 10a shows the initial classification of bright/dark/clutter for the test pattern. A solid section of double "DD" characters exists in the dark center of the octagon; and an undecided area represented by single "D" characters surrounds the solid section. Most of the outer bright area of the octagon is indicated by double "BB" characters, with single "B"s on the inner edge and near the darker corners. The area between the bright and dark region is classified with probabilities of being clutter between 0.3 and 0.5. The upper and lower left corners are represented by double "DD" characters; however, the lower left corner is somewhat larger.

Iteration 1: The undecided regions are reduced; the principal effect noticed is the higher probabilities for the clutter area between the inner dark region and the bright outer ring as indicated by the double "CC" characters.

Iteration 2: The undecided regions are nearly eliminated leaving the solid sections of double "BB", "CC", and "DD".

Iterations 3 - 13: The relaxation process has driven the probabilities toward unity for the appropriate label for that region (see Figure 10d-e).

Iteration 100: Virtually no change from iteration 13; the only effect has been some smoothing of boundaries, especially at corners. It may be noted that the sizes and shapes of the three prinicpal regions are basically the same as originally defined at iteration 0 and 1. Actually, the dark central square corresponds exactly to the region of input

gray levels equal to .8 or greater, despite some size increase during the intermediate iterations.

B. Danker's Window, Figure 11

Iteration 0: Figure 11a shows that the window is covered mostly with double "BB" characters except for the target blob represented by a group of blanks and random noise represented by single "B" characters. Figure 11b shows a small blob with single "D" characters (note that this is processed with the convention that large gray values are dark). Figure 11c shows that the window is covered mostly with single "C" characters indicating a slight chance of being classified as clutter ($0.3 < P_iC < 0.5$).

Iteration 1: The relaxation process has driven the undecided regions toward either the bright or the clutter label. The size of the undecided clutter region has been reduced rapidly except for the area where the blob is located; at the same time, the probabilities for the pixels within the blob were driven up toward clutter (see Figure 11e). The blob of single "D" characters has been nearly eliminated by the relaxation process (see Figure 11f). The reason the target blob is not labeled dark can be seen from the histogram, which shows that the maximum value is 47, which is only .75 of full scale.

Iteration 8: Figures 11g-h show convergence of probabilities for all pixels in the window toward either the bright or the clutter label.

Iteration 100: Figures 11i-j show the stability of the relaxation process. The shape of the blob in both figures has changed only slightly from iteration 8; the size of the blob has remained essentially the same.

It is of interest to compare these results with the comparable two-label process results of Figure 4. The final results (iteration 8 or 100) are virtually identical with complete segmentation of the target blob. The only difference is that the equivalent bright/clutter threshold is lower for three label relaxation, so the target blob is slightly larger; likewise, the clutter region in the lower left corner is larger so it shrinks more slowly. The clutter

printout of the first iteration of Figure 11 compares very closely to the initial printout of Figure 4.

C. <u>MODPOT 26 Window, Figure 12</u>

The MODPOT 26 window is a Westinghouse test pattern, originally containing two bright target blobs of FLIR imagery from the NVL data base. The window also contains defects in the form of bright streaks at the bottom of the frame. Three dark blobs have been superimposed onto the background by adding fixed values to form a circle, a diamond, and a smaller square. The diamond is not symmetrical (the upper left and the lower right sides are longer) and the upper left side has a "hump" with a square corner. The image shown in Figure 12 has been filtered by one stage of 3x3 cascaded median filtering followed by one stage of 3x3 weighted filtering.

<u>Iteration 0</u>: The bright printout shows two bright blobs represented by double "BB" characters and some shadow areas surrounding the blobs shown with single "B" characters. The rest of the window is covered with blanks except for the lower streaked area also represented by double "BB" characters. The dark printout shows a circle, a diamond (with the hump), and a square represented by "DD" characters. The clutter printout for the window is shown with single "C" characters except for the bright and dark blobs, and part of the lower streaked area.

In summary, the original bright blobs and streaks in the window were labeled as bright; the dark figures were labeled as dark; and the background was undecided.

<u>Iteration 1</u>: The bright and dark areas are better defined. The background which was undecided before is converging to the clutter label. The shape of the geometric figures has not changed; especially notice that the hump on the diamond B is still intact.

<u>Iteration 2, 4, and 8</u>: During these iterations, the relaxation labels have become very well defined. The background of the window has converged to the clutter label. The shape of the circle has not changed,

while the hump and corners of the diamond have been rounded; also the corners of the square have been rounded. It is noted that the lower bright blob is merging with the bright streaked area immediately below it.

Iteration 100: This shows again the stability of the relaxation process. The shape of the blobs and the geometric figures stays essentially unchanged from iteration 8. The bright area remains as bright, while the dark and clutter areas remain the same respectively. The size of the objects remains virtually unchanged from the initial and first iterations.

The three label relaxation process has been demonstrated on three types of image data, which may be categorized as follows:

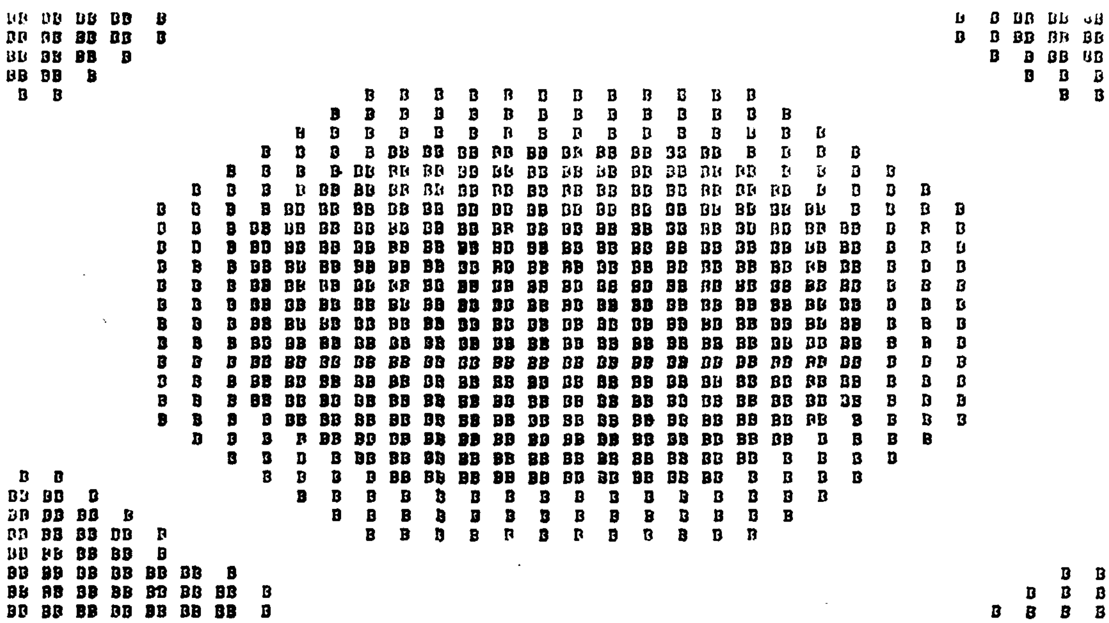
1. A continuously varying gray level in the octagon test pattern (Figure 10).
2. An image with two principal subpopulations (Figure 11).
3. An image containing two distinct target subpopulations and an intermediate background subpopulation (Figure 12).

The results shown in Figure 10 are rather interesting. Although the three label process was used with no compensation for threshold differences, the segmentation quickly converged to two labels with virtually the same results as the successful results with two label AUTO-R processing. This leads to the conclusion that AUTO-R processing with multiple labels is rather robust: It is not very sensitive to threshold adjustment, tolerates more labels in the process than the number of subpopulations in the image, and converges quickly to a stable result that segments the regions in the input image.

The three label results for both the continuously varying octagon test pattern and the MODPOT 26 window with three subpopulations were both very satisfactory. The second iteration resolved most of the ambiguity and segmented the image into regions that accurately represented three levels in the input image. Because the process is stable, further processing had little effect to provide further definition and smooth boundaries.

41

FILE NAME= TESTP2.DAT
BRIGHT/DARK/CLUTTER RELAZATION 32 BY 32
KB= 1.00KBC= 0.00
KD= 0.00KCD= 0.00
KC= 1.00KBCD= 0.00
DELTA= 0.00
NUM= 1
[illegible]=0.B1B=1.00
BRIGHT



CLUTTER

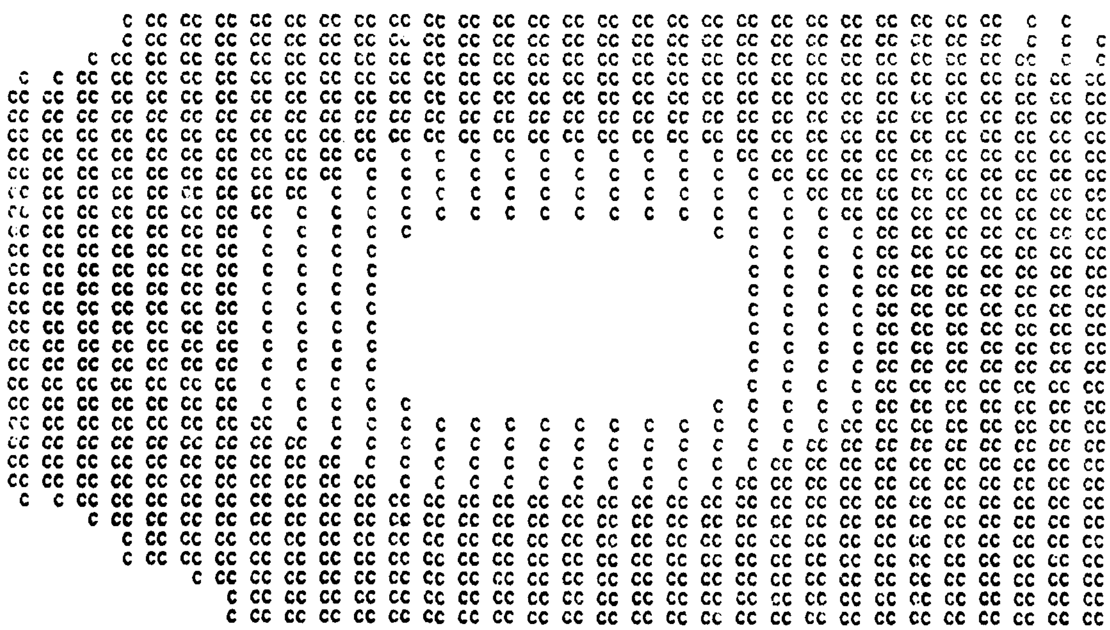


Figure 1a

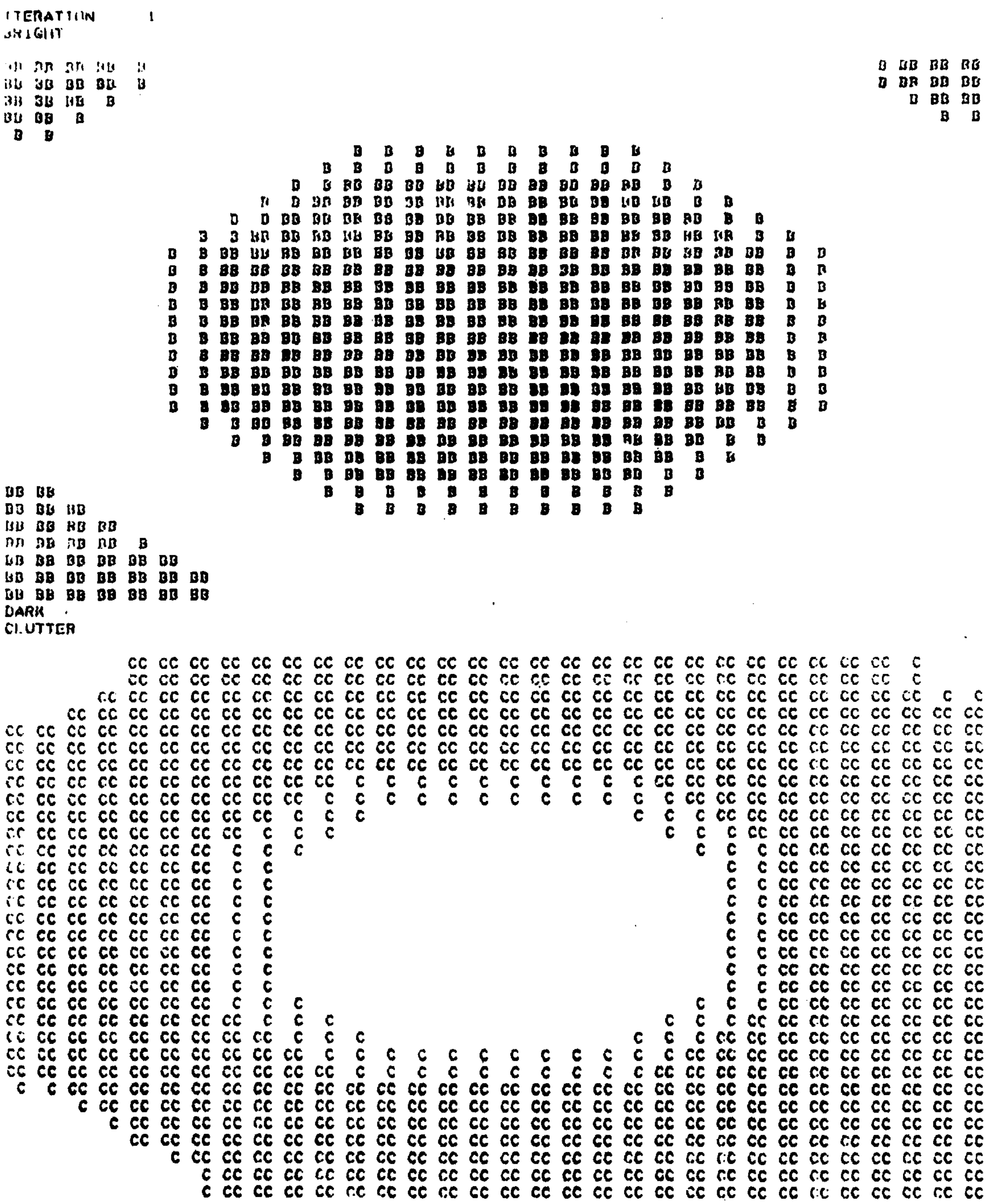


Figure 1b

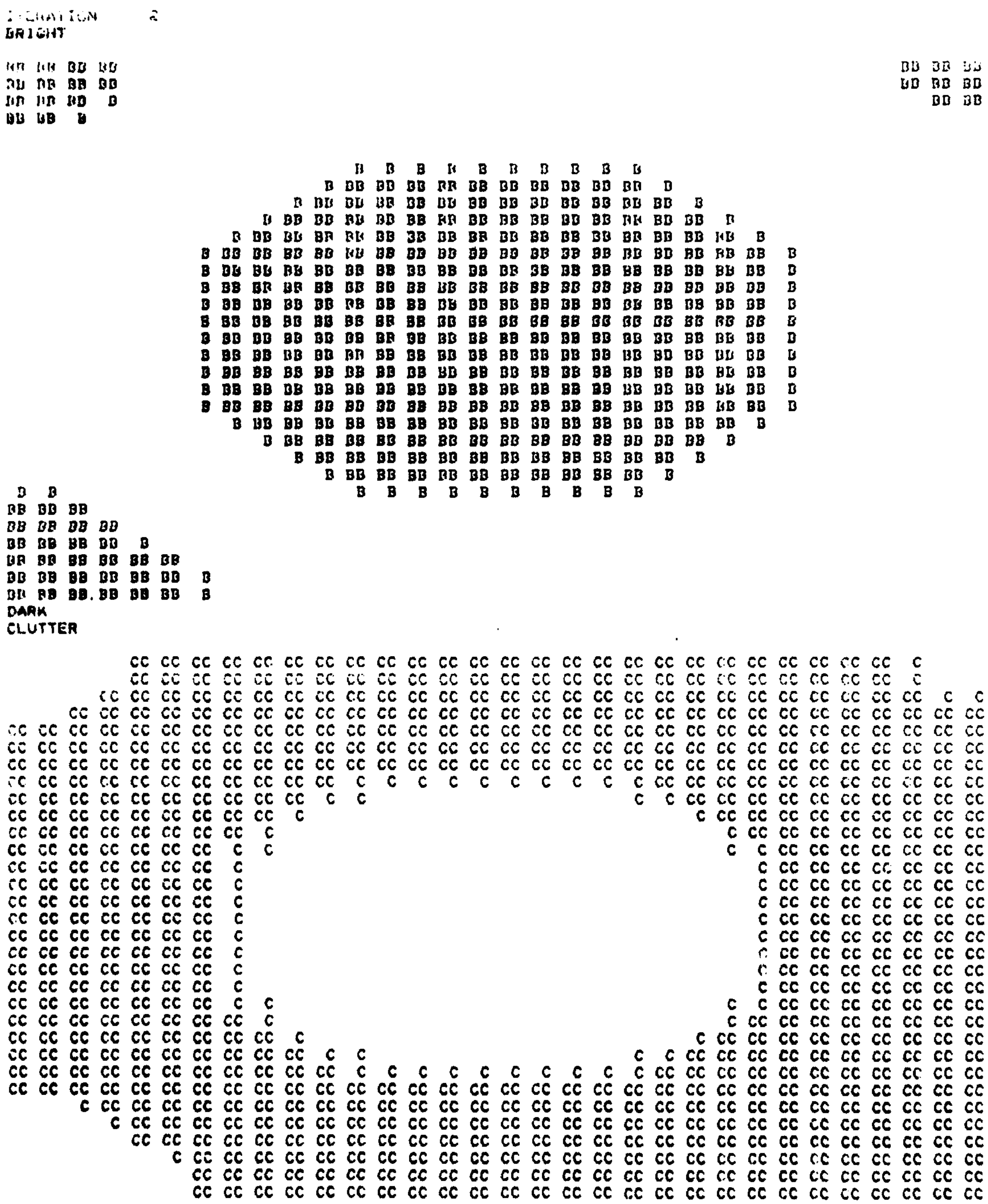


Figure 1c

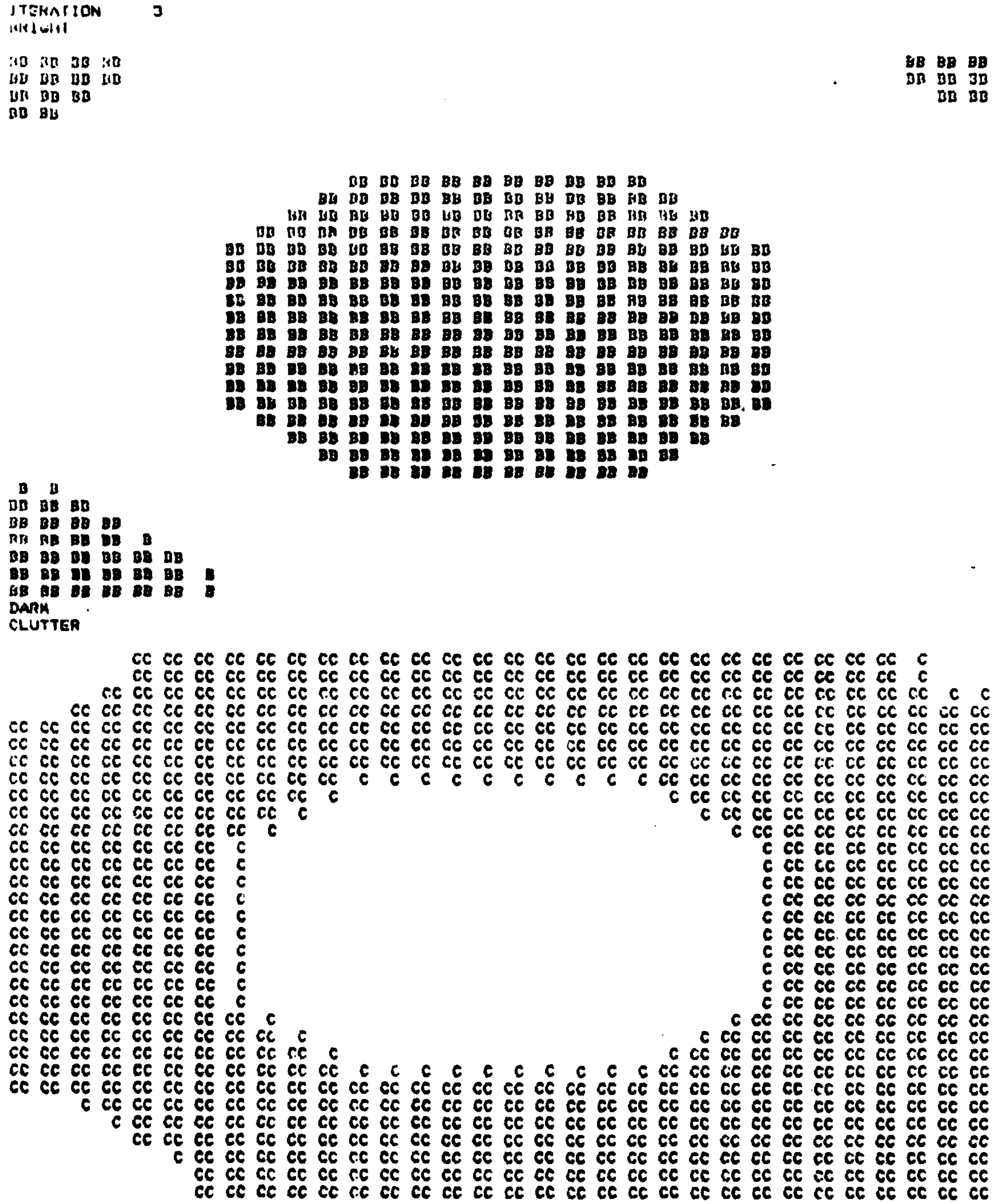


Figure 14

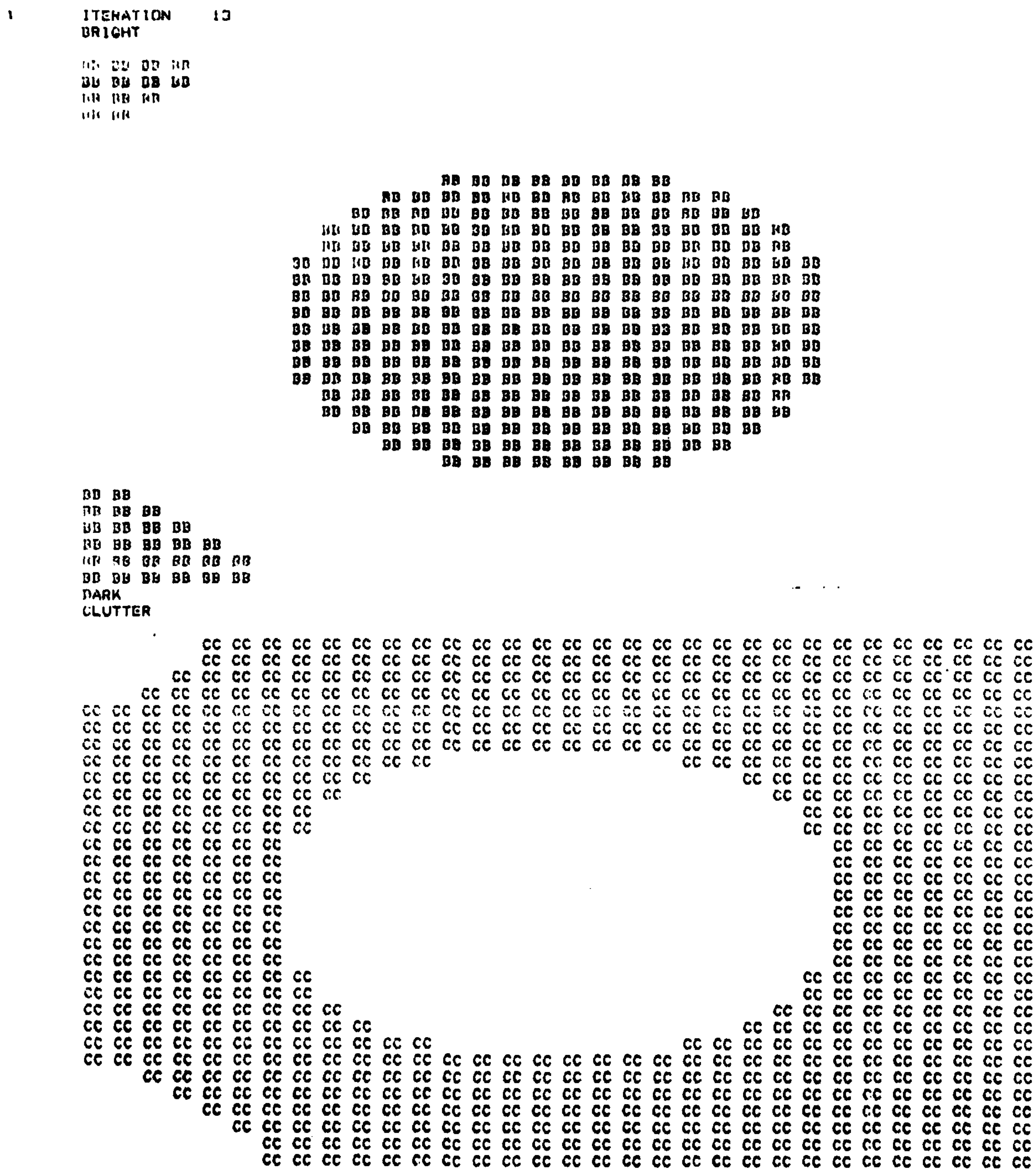
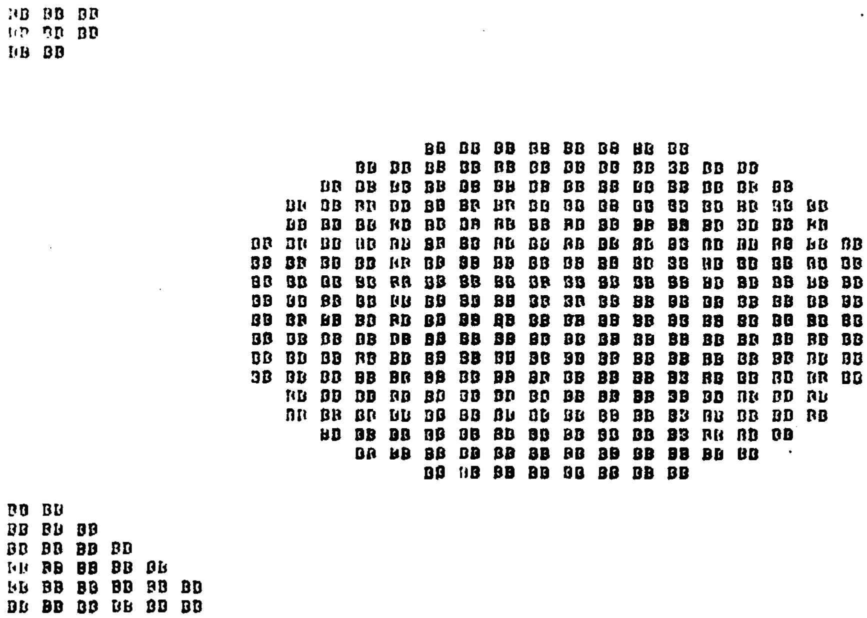


Figure 1e

ITERATION 100
BRIGHT



DARK
CLUTTER

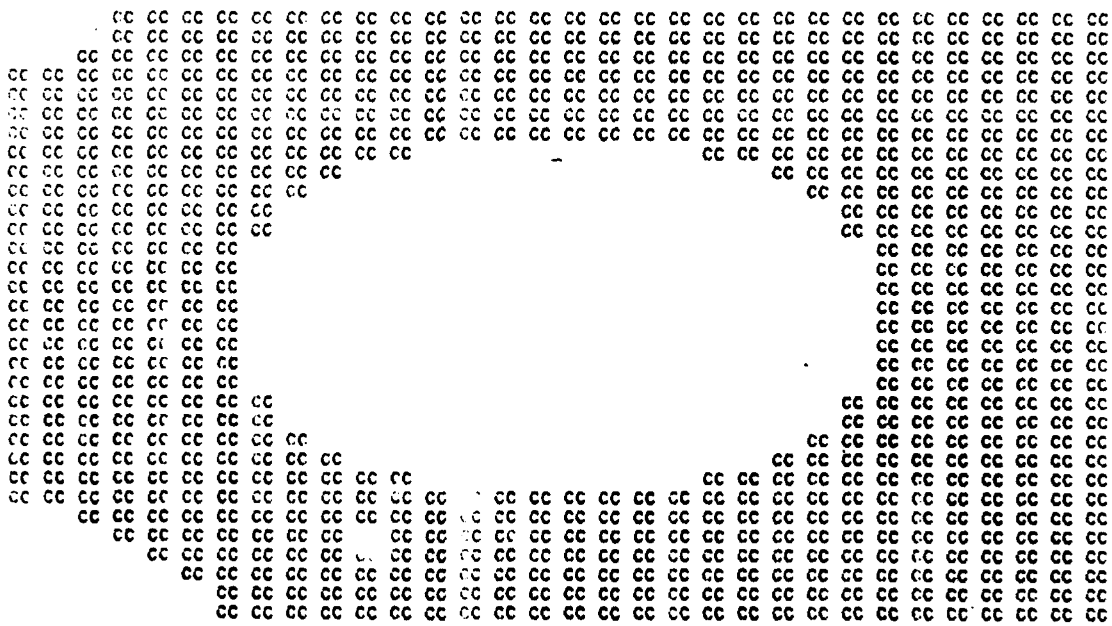
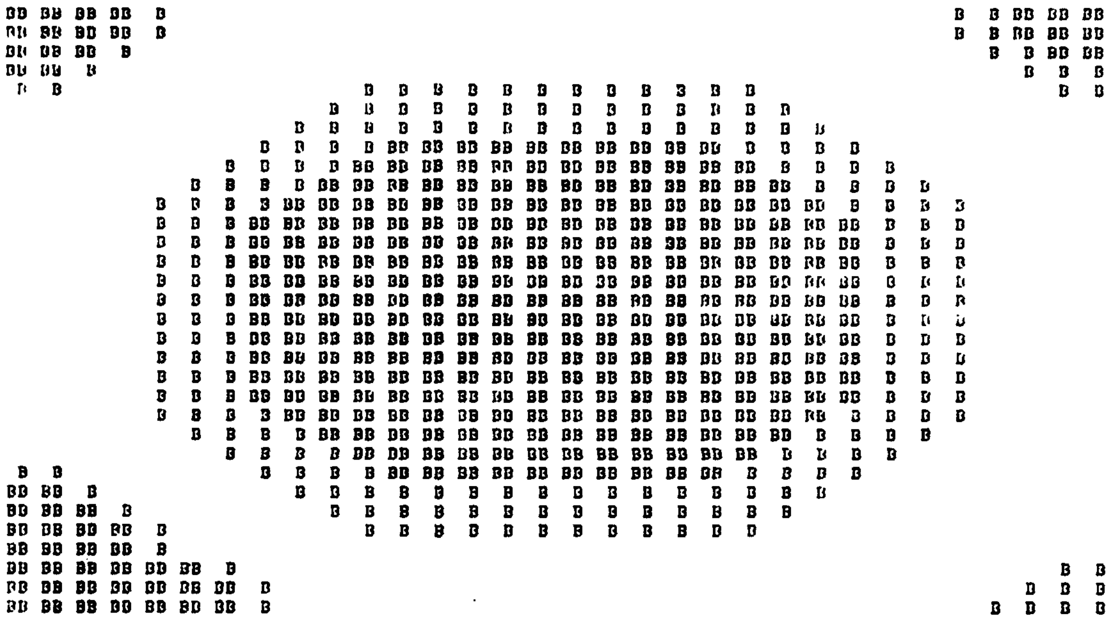


Figure 1f

```
FILE NAME= TESTP2.DAT
BRIGHT/DARK/CLUTTER RELAZATION 32 BY 32
KB= 0.10KBC= 9.00
KD= 0.00KCD= 0.00
KC= 0.10KBCD= 9.00
DELTA= 0.00
NUM= 1
FB=0.DIB=0.BIB=1.00
BRIGHT
```



```
CLUTTER
```

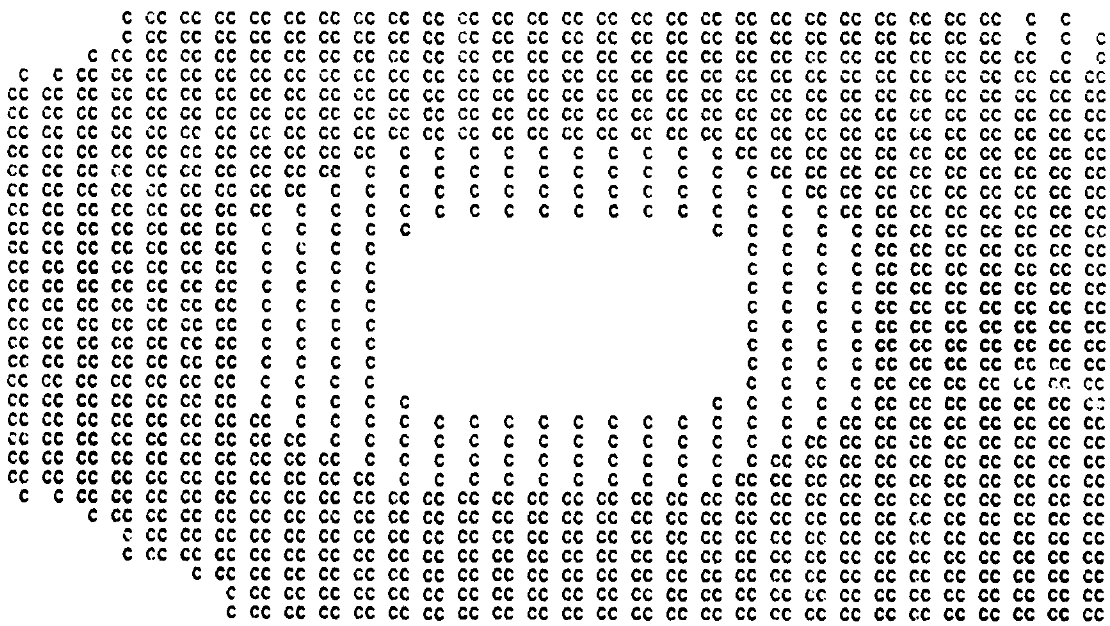


Figure 2a

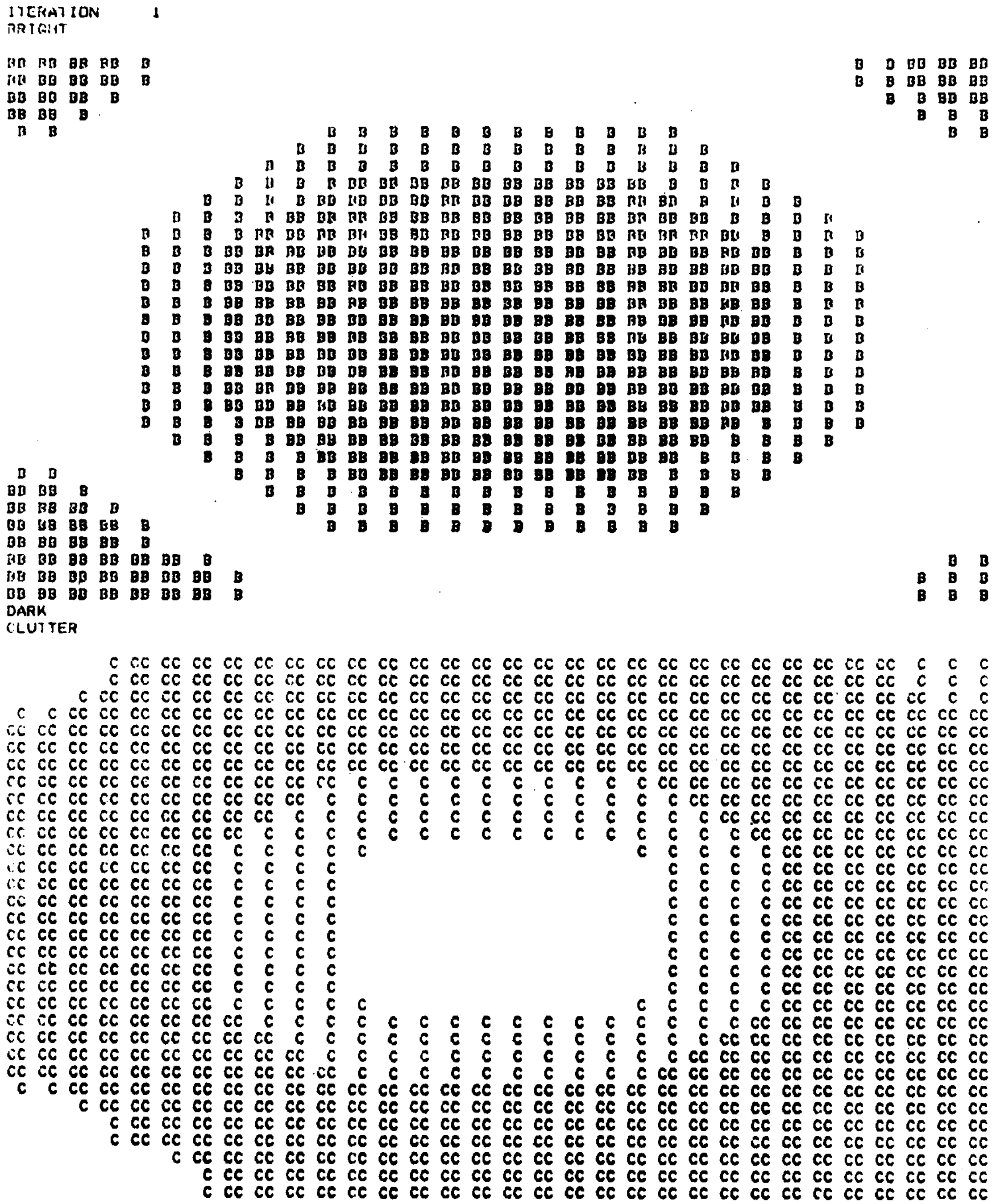


Figure 2b

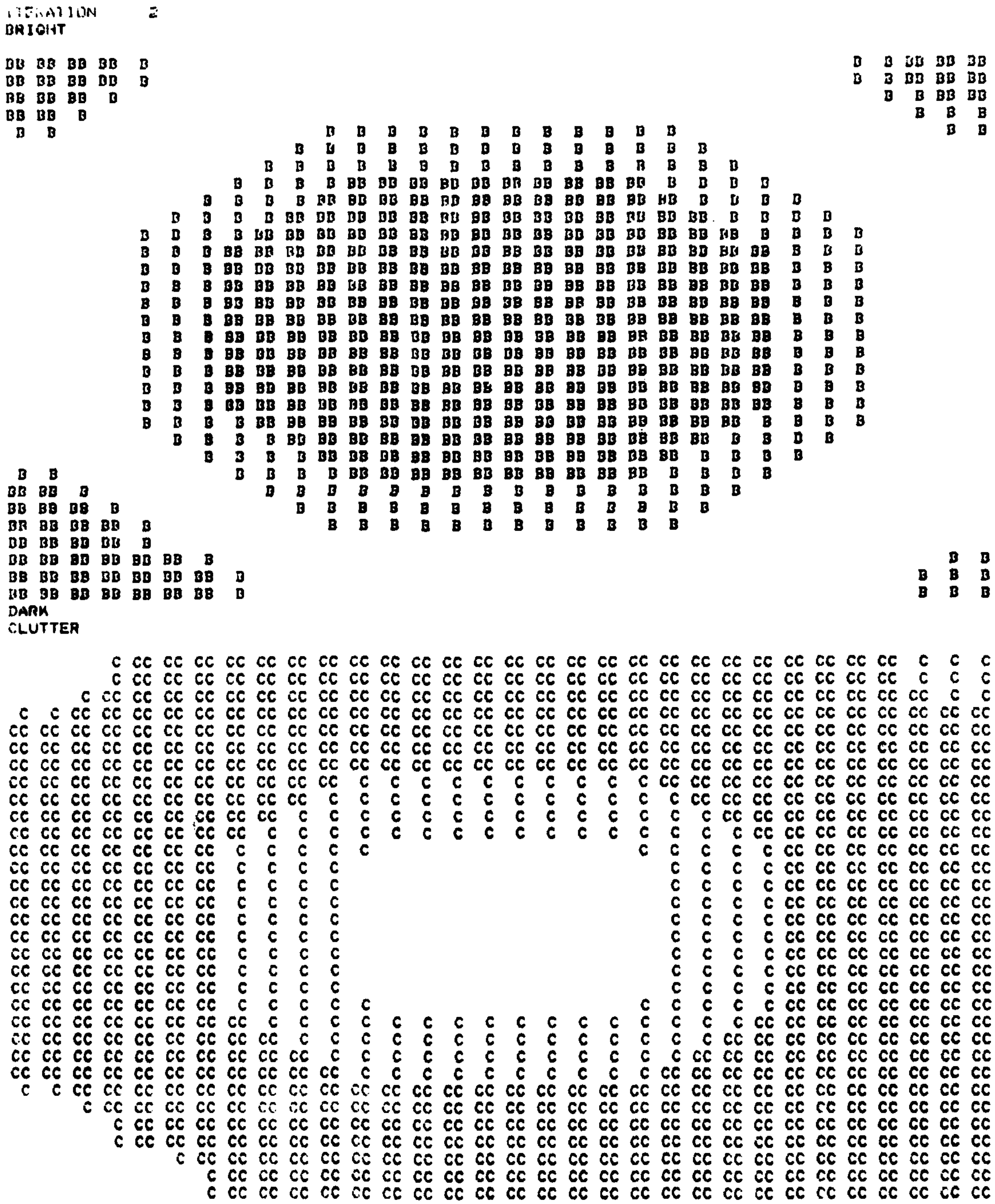


Figure 2c

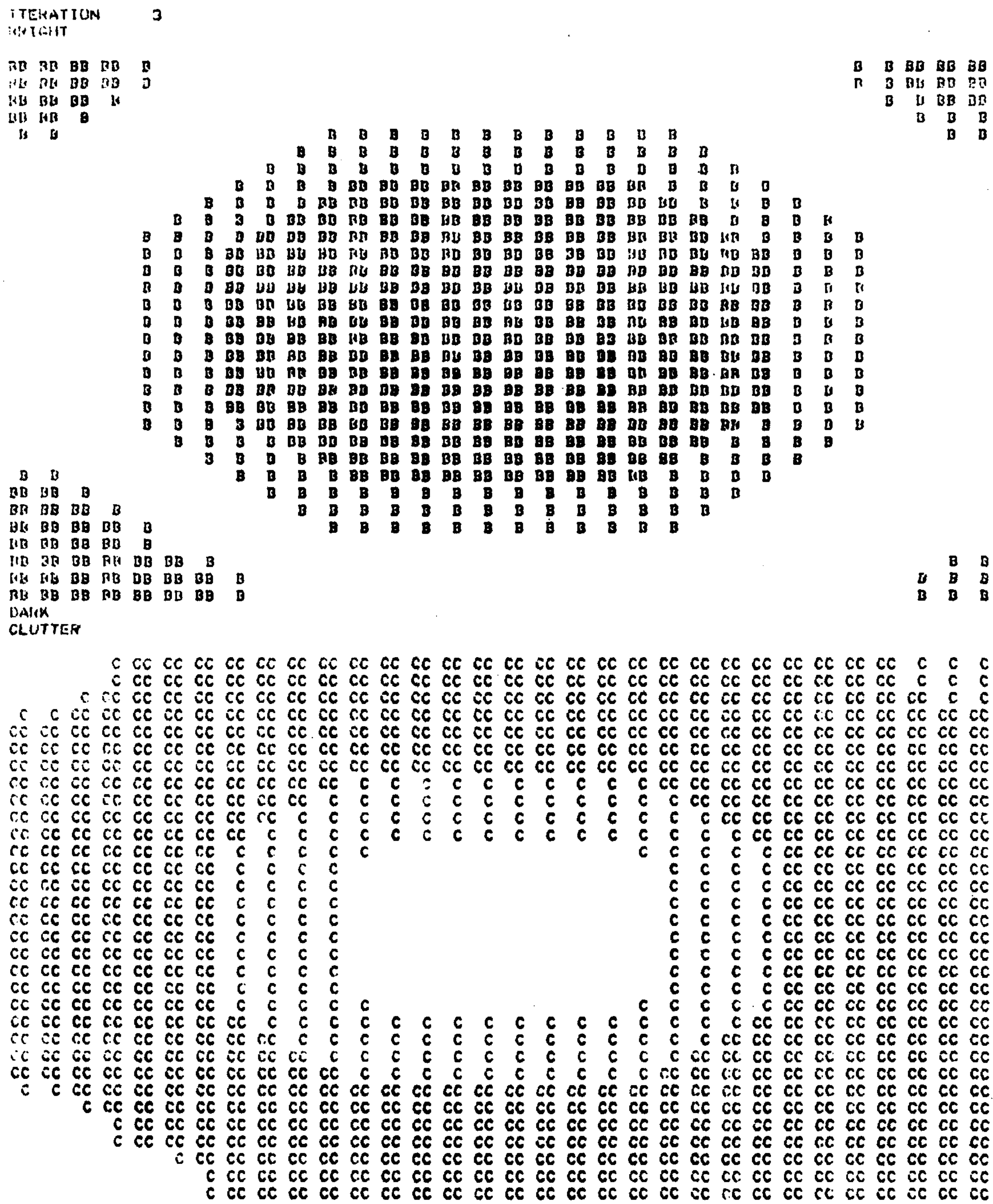


Figure 2d

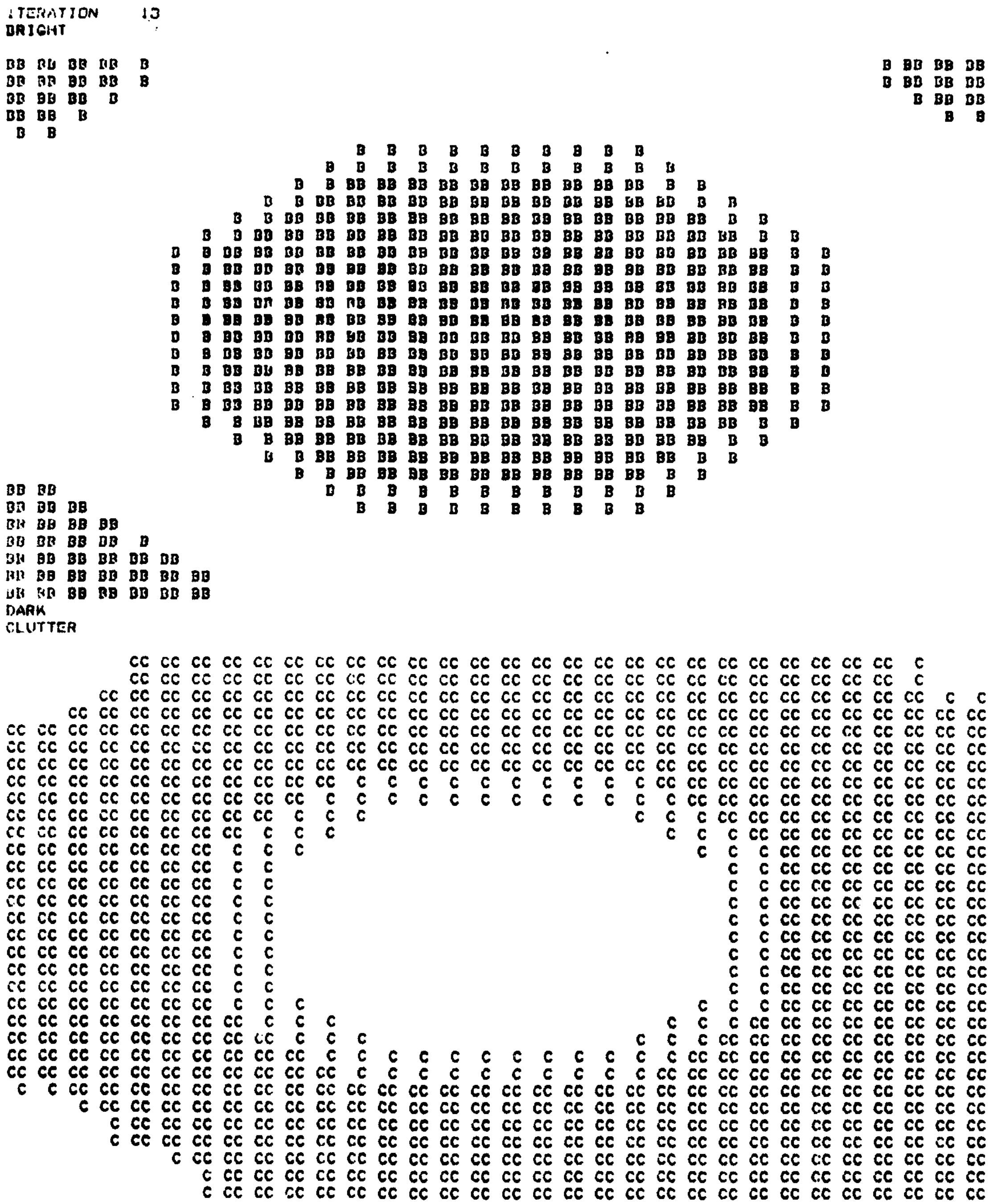


Figure 2e

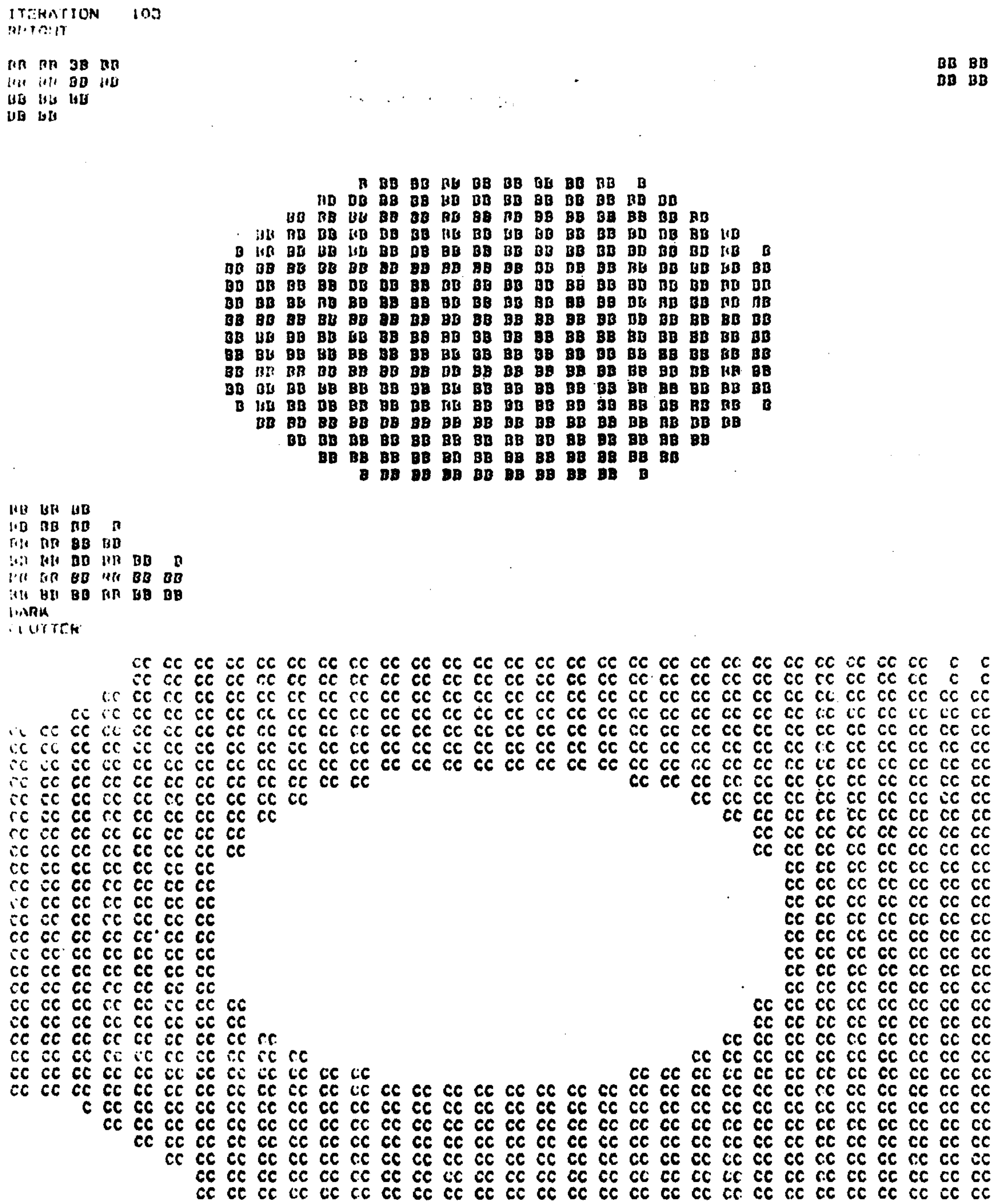


Figure 2f

FILE NAME= TESTP2.DAT
BRIGHT/DARK/CLUTTER RELAZATION 32 BY 32
KB= 1.07KBC= 0.78
KD= 0.00KCD= 0.00
KC= 0.[illegible]KBCD= 4.15
DELTA= [illegible] 00
NUM= 1
FRG. DIB=0. BIB=1.0

ITERATION 0

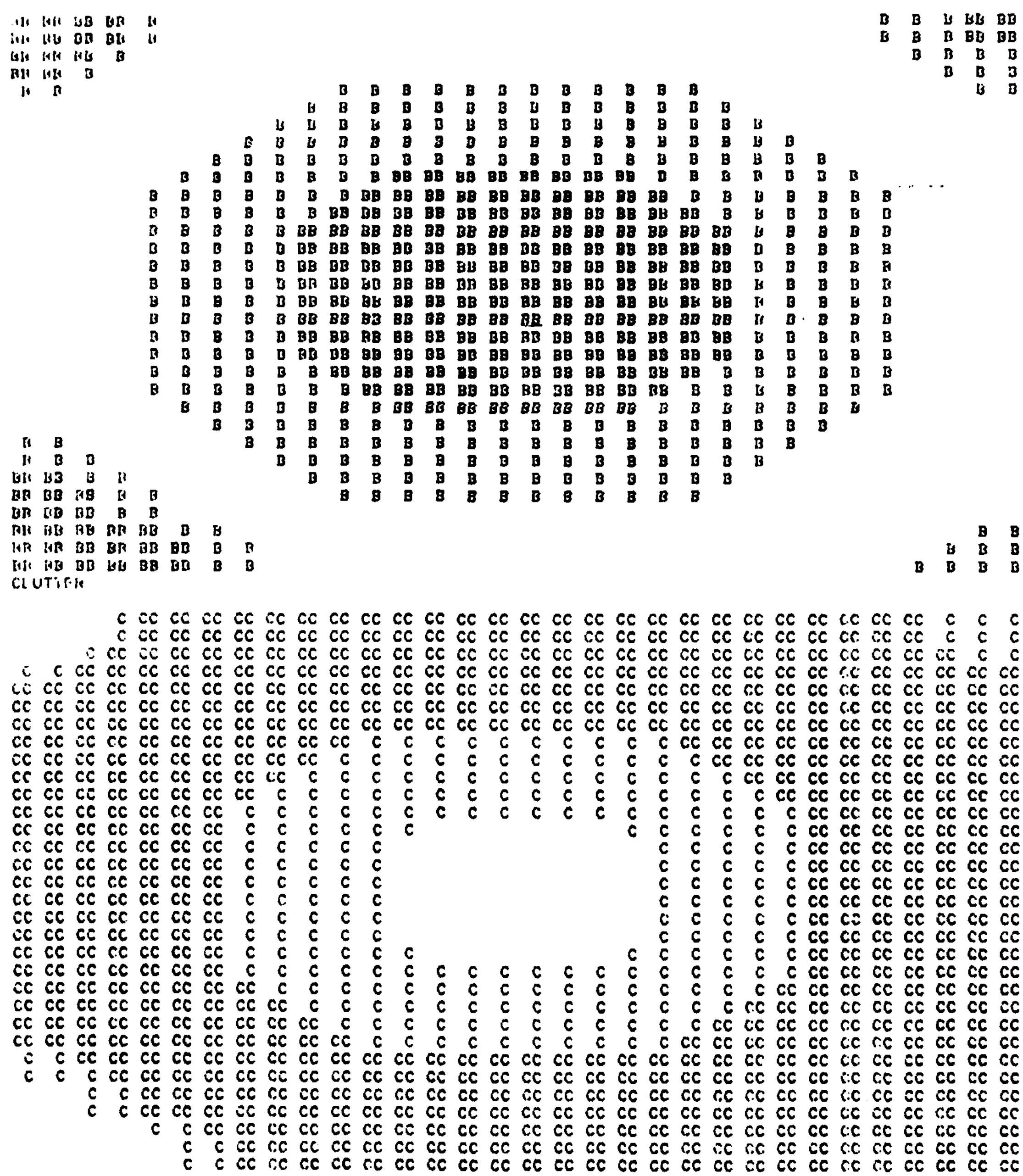


Figure 3a

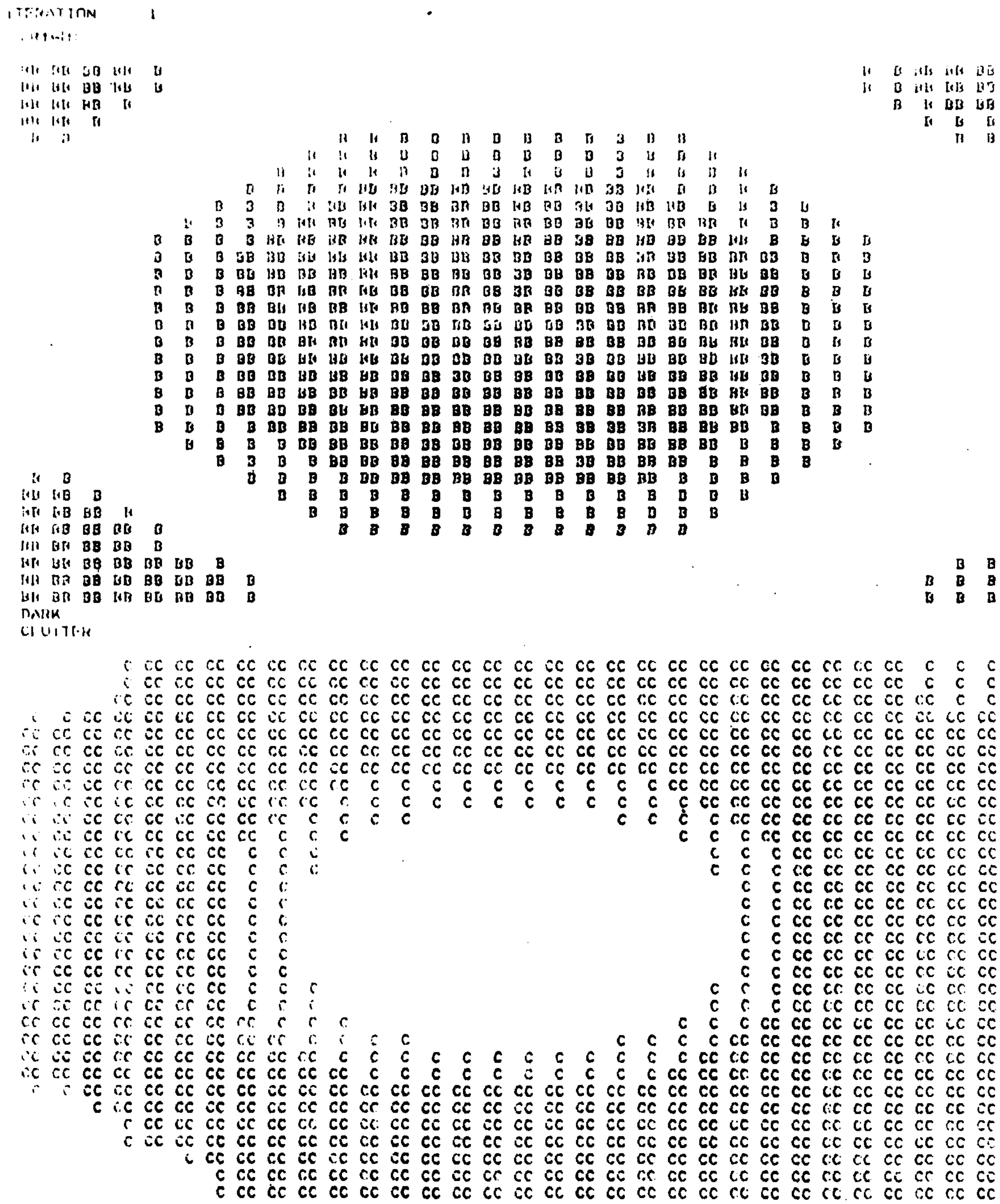


Figure 3b

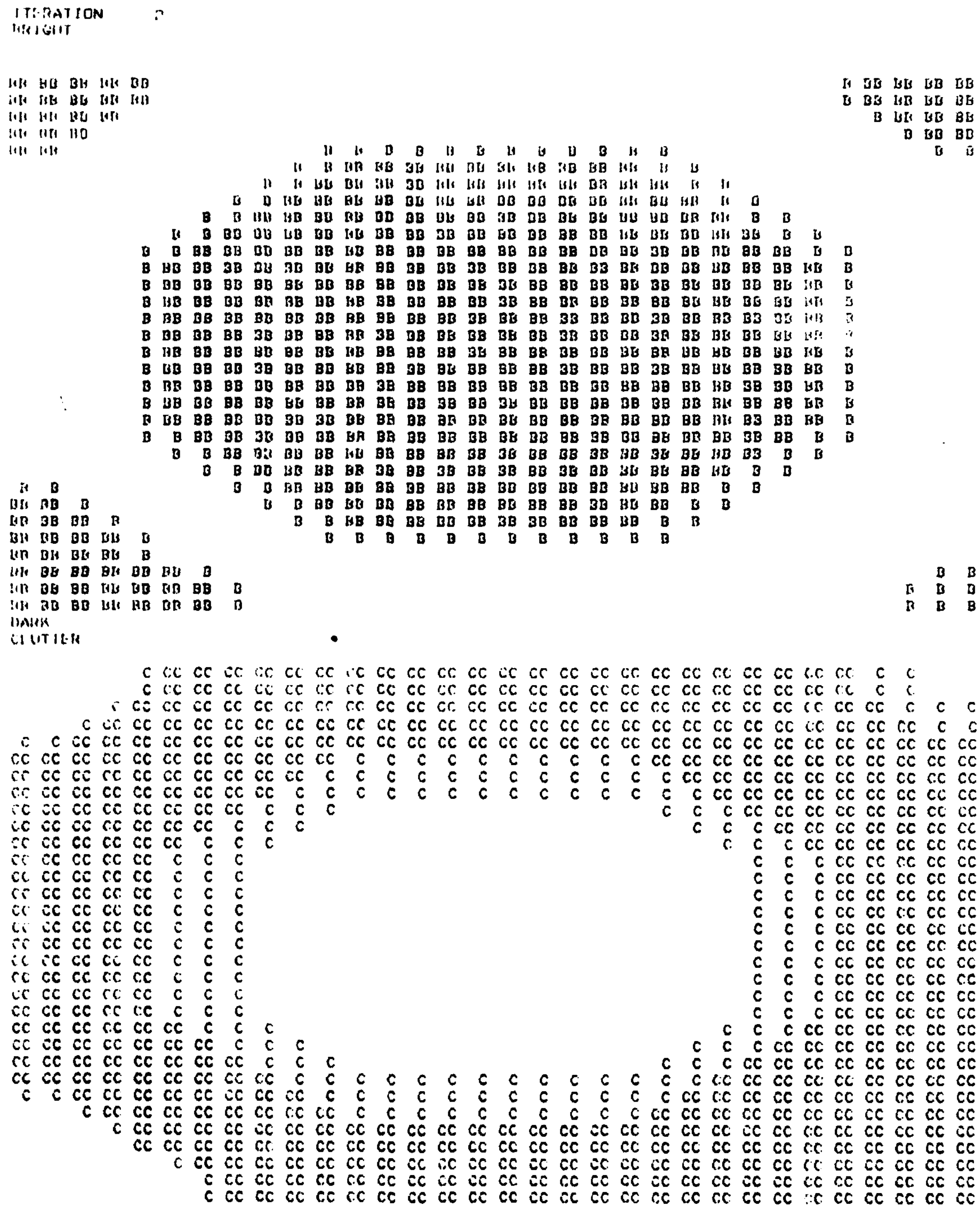


Figure 3c

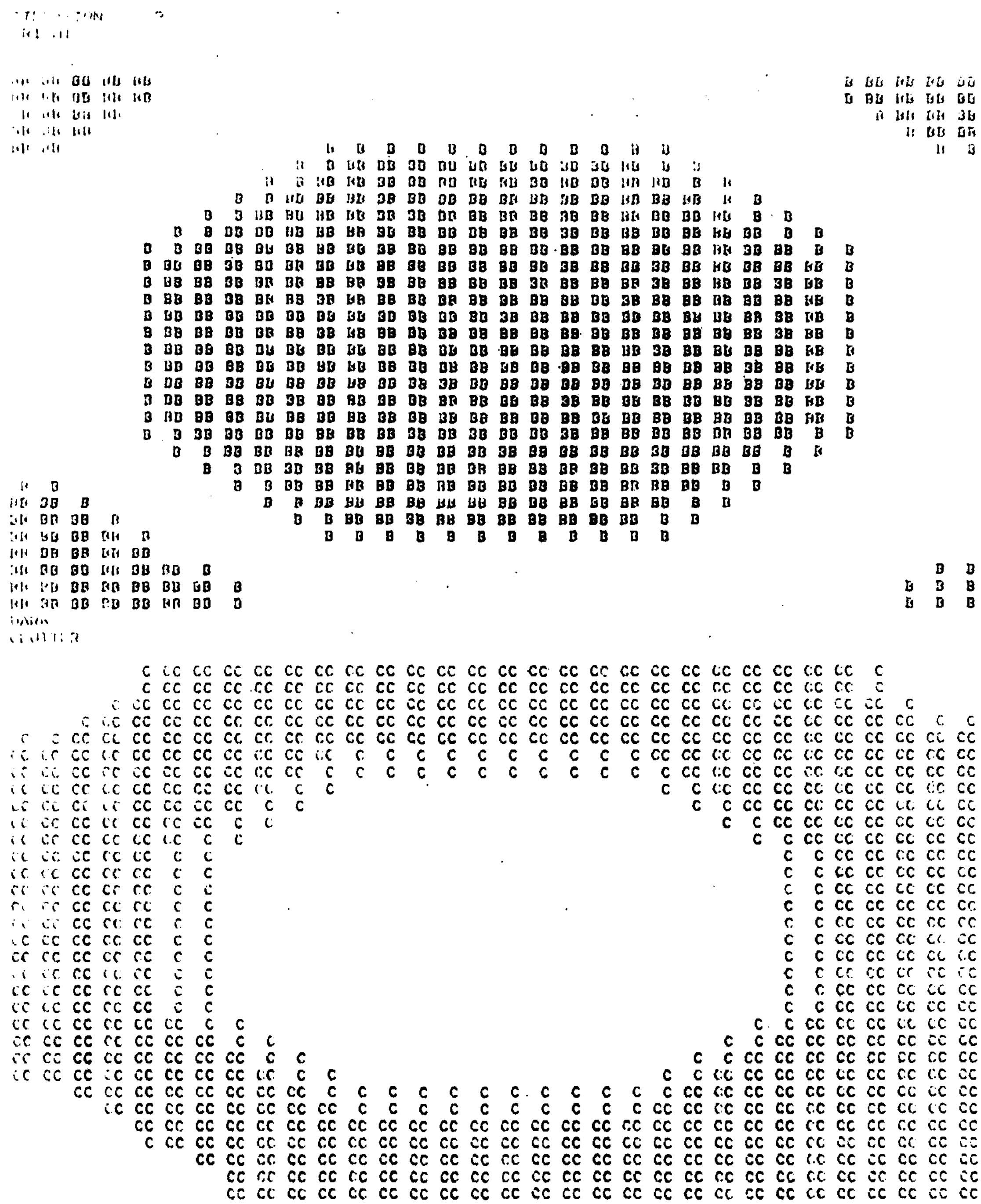
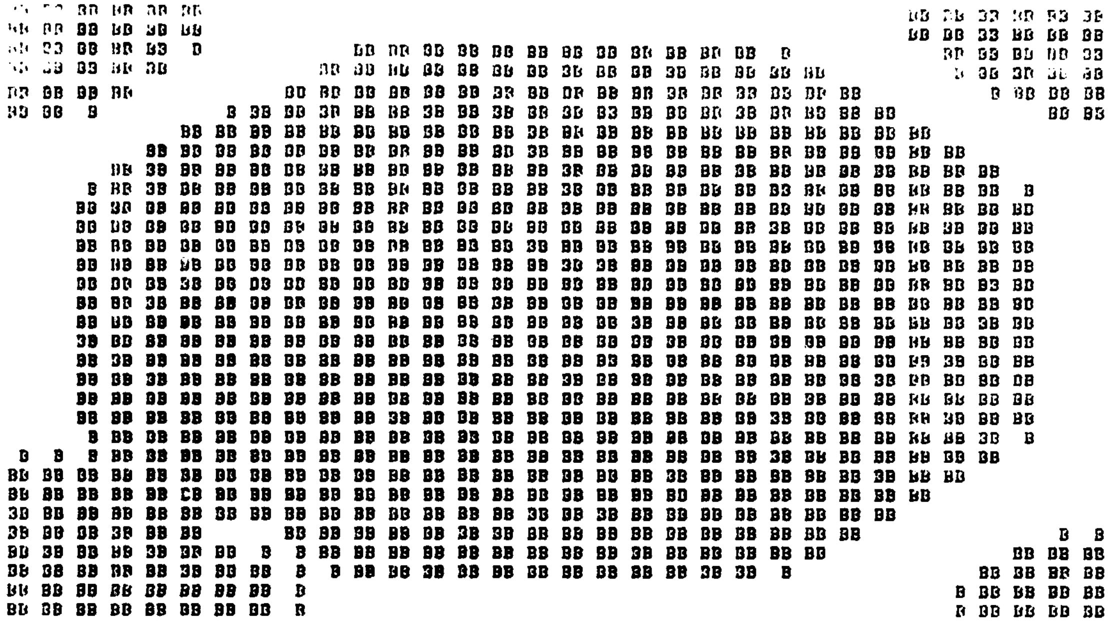


Figure 3d

ITERATION 13
BRIGHT



DARK
CLUTTER

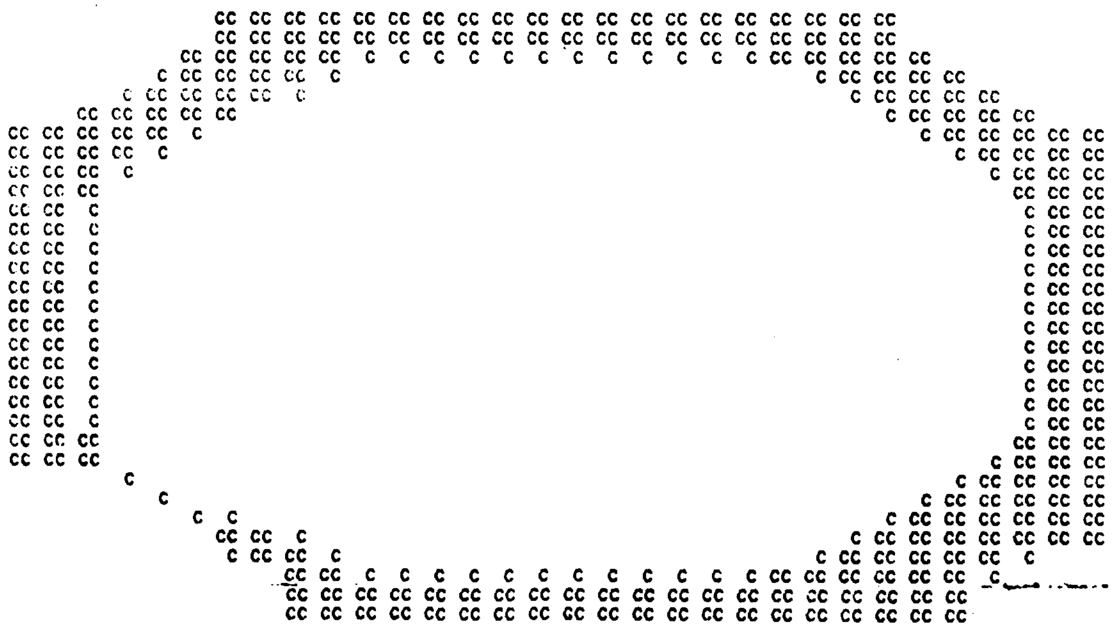
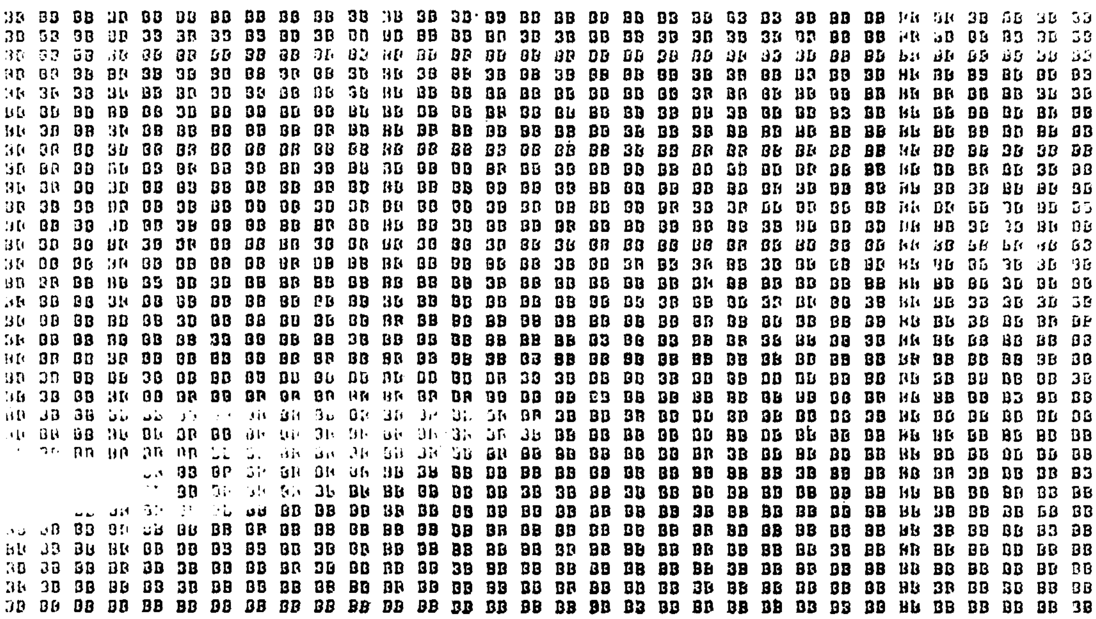


Figure 3e

ITERATION 100

BRIGHT



DARK
CLUTTER

Figure 3f

FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.00KBC= 0.00
KD= 0.00KCD= 0.00
KC= 1.00KBCD= 0.00
DELTA= 0.00
FB=G .DIB=0,BIB=1.00

Figure 4

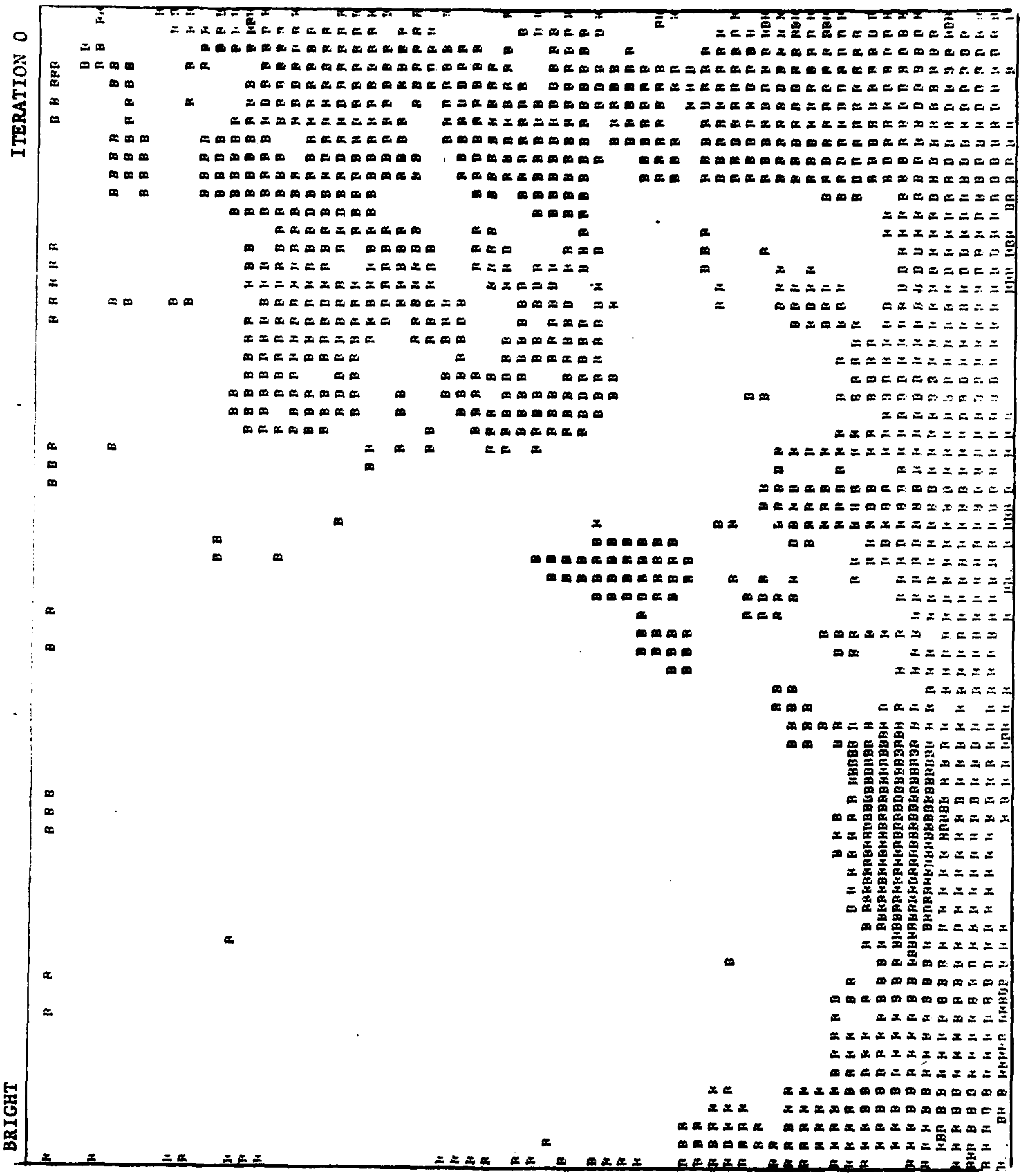


Figure 4a

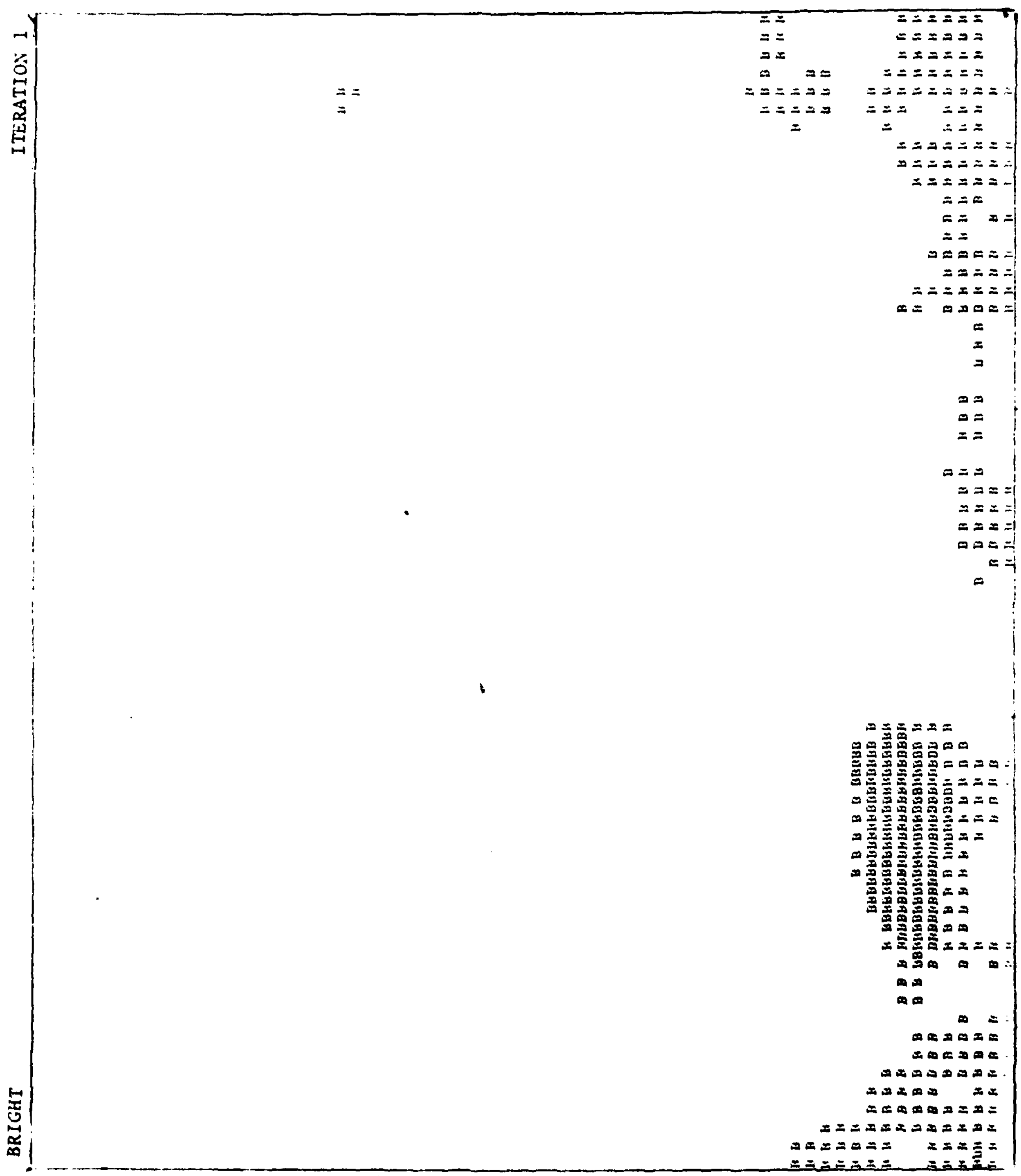


Figure 4b

BRIGHT ITERATION 2

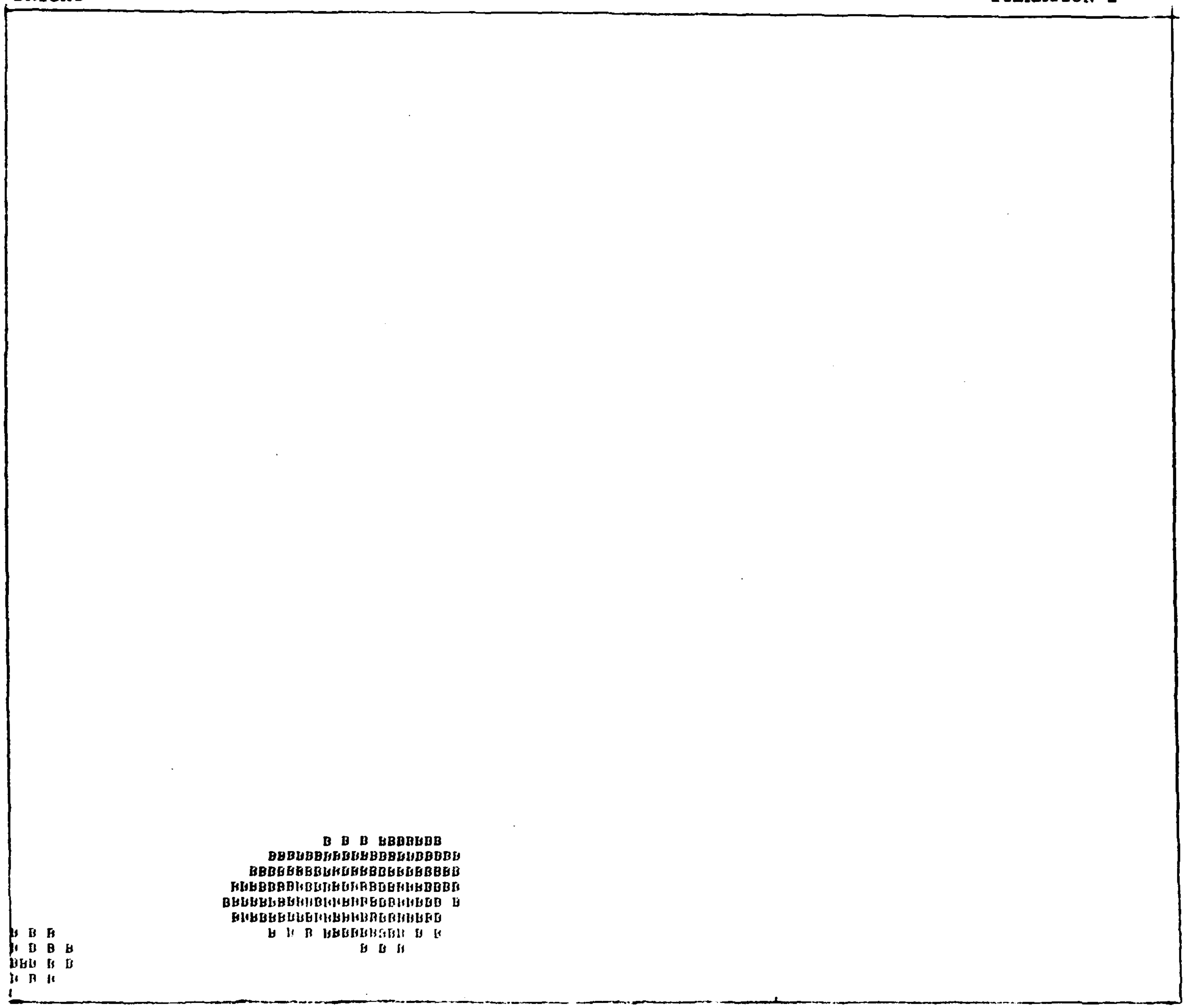


Figure 4c

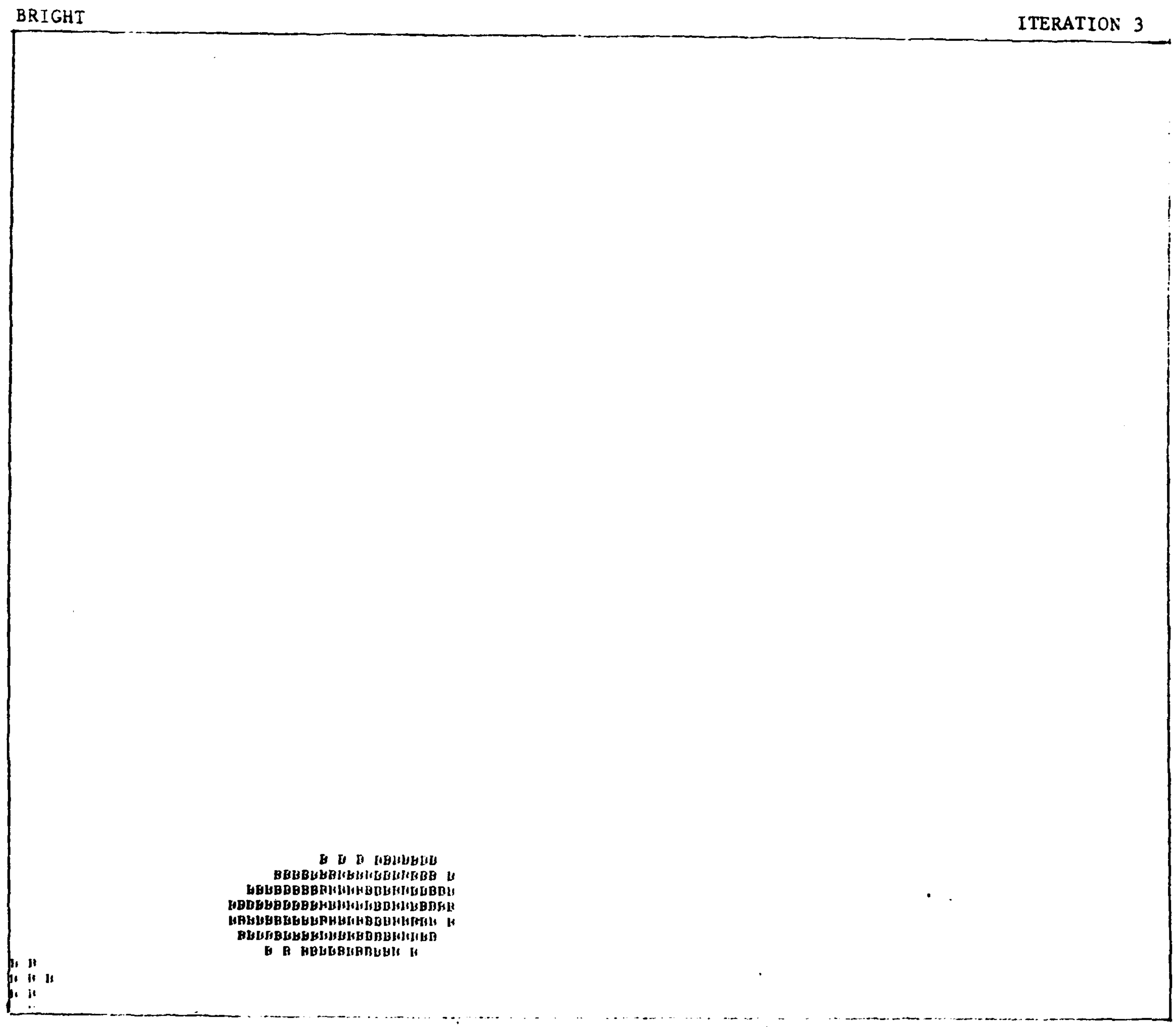


Figure 4d

BRIGHT

ITERATION 4

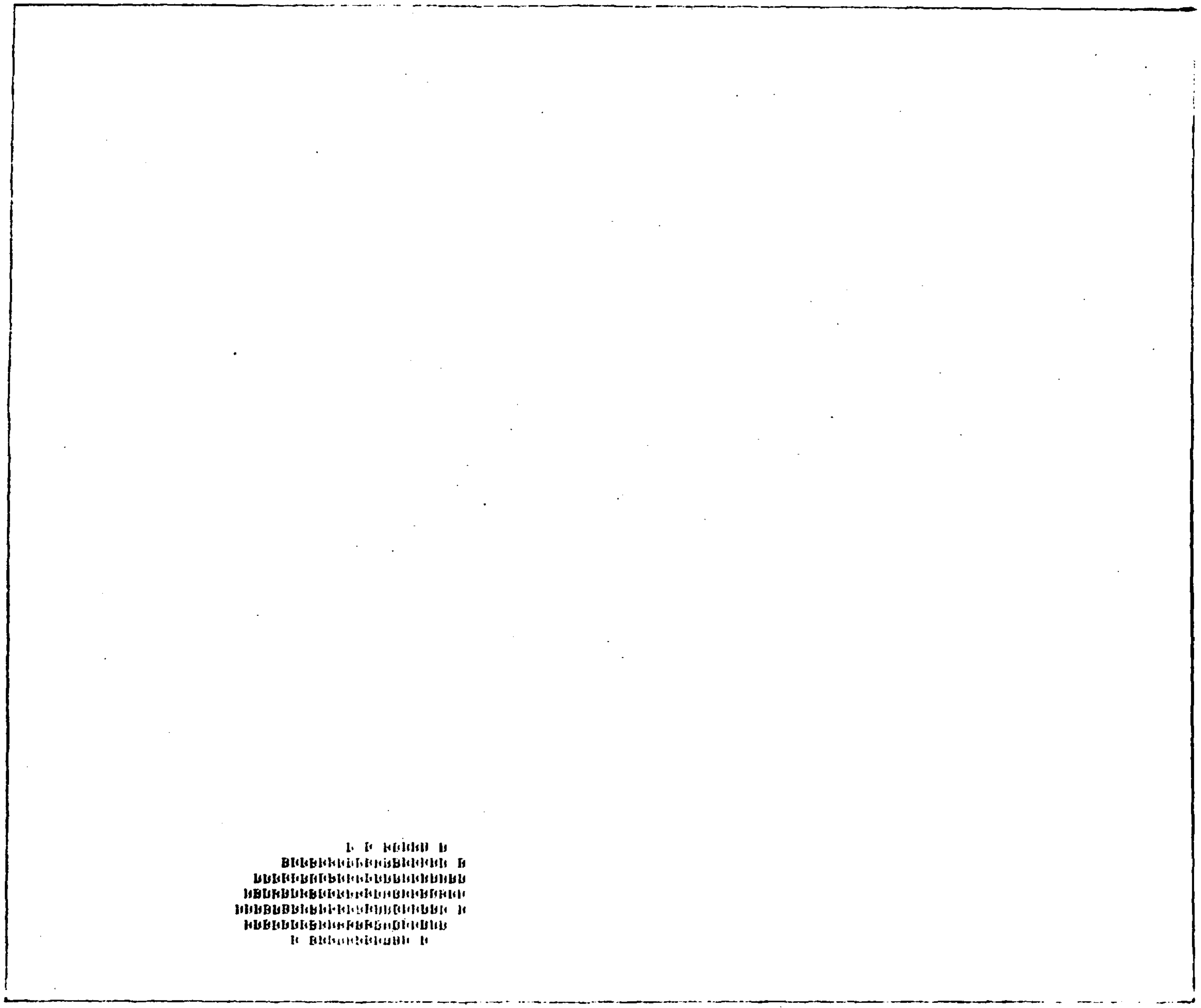


Figure 4e

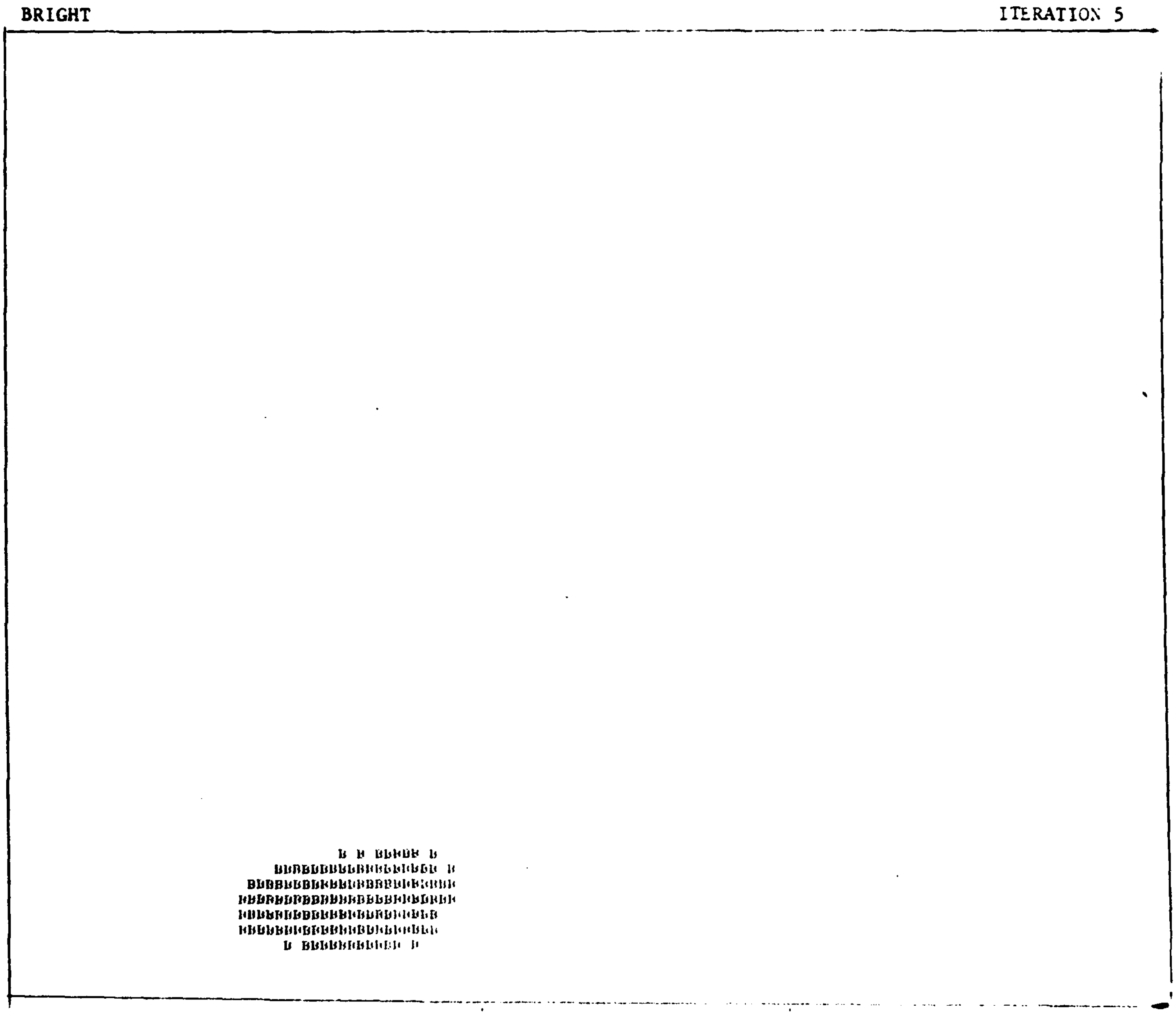


Figure 4f

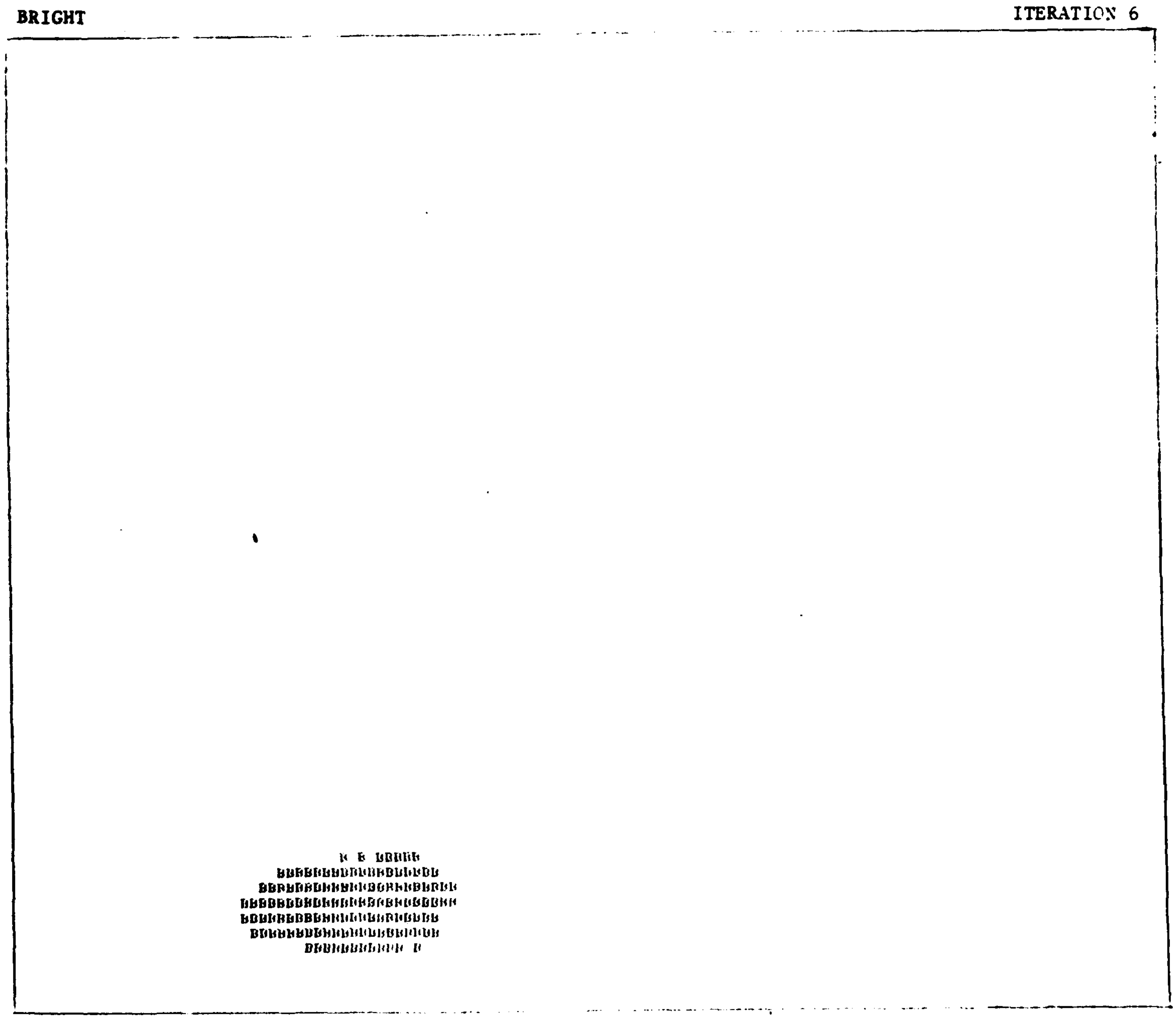


Figure 4g

BRIGHT

ITERATION 7

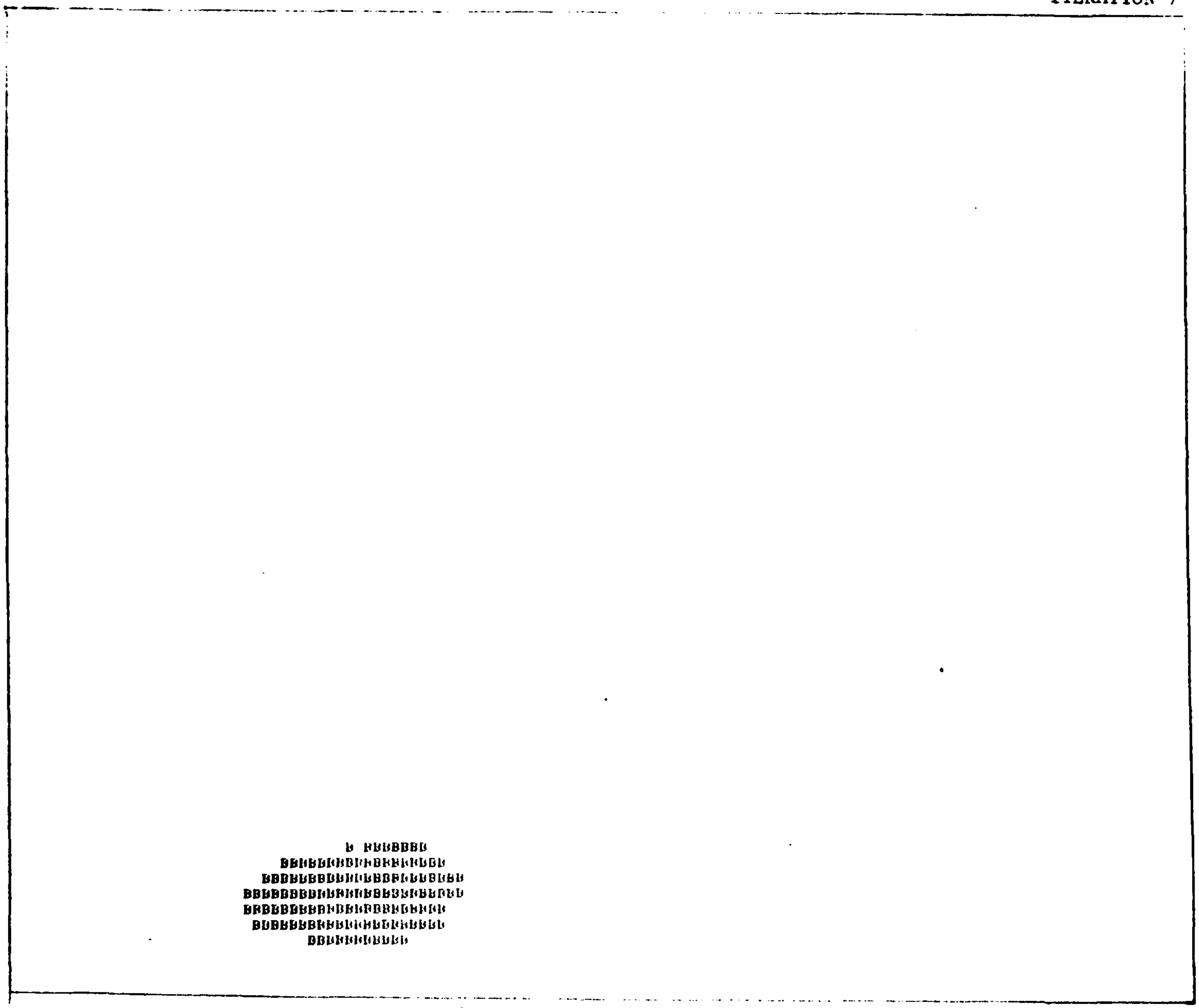


Figure 4h

BRIGHT ITERATION 8

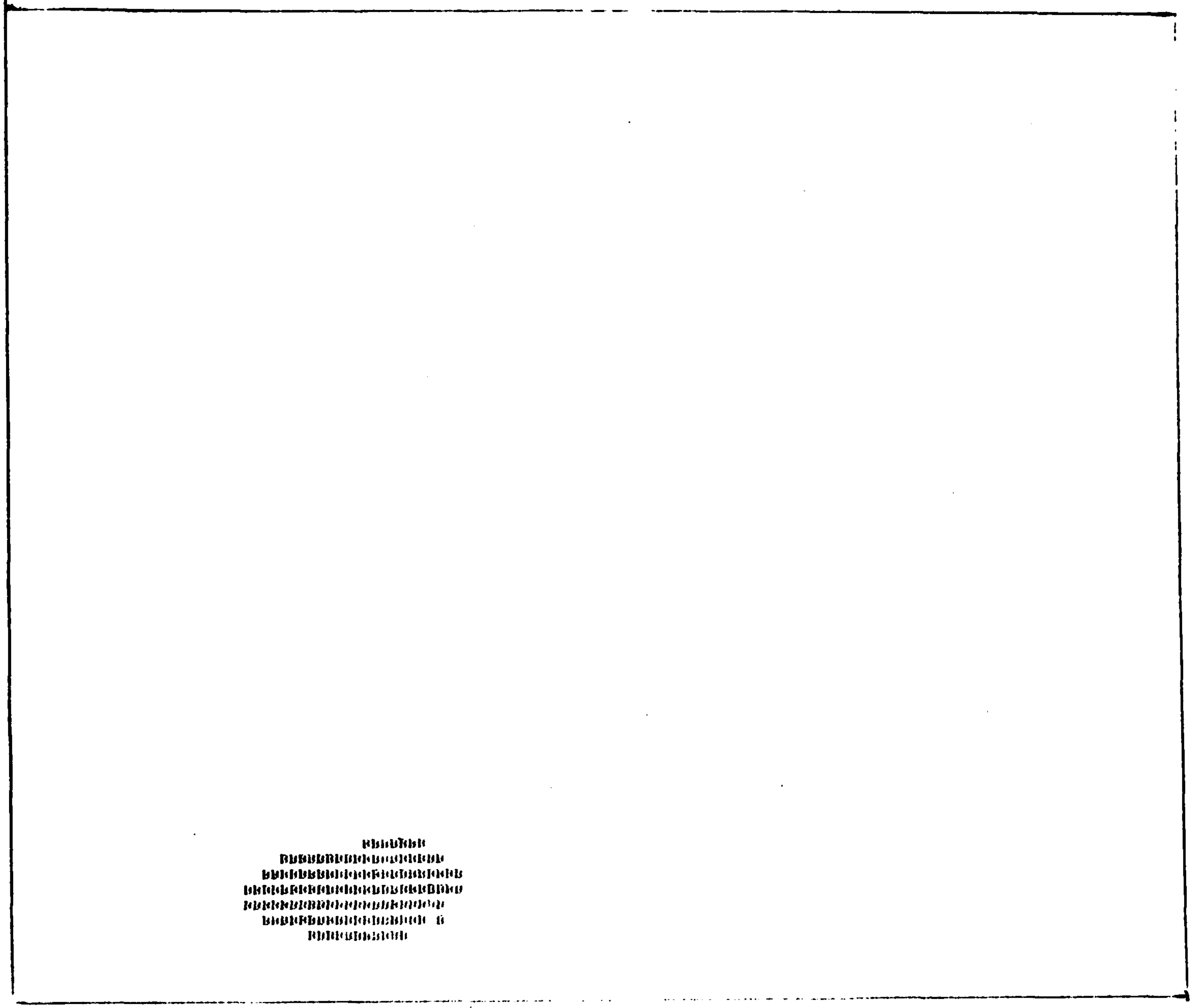


Figure 4i

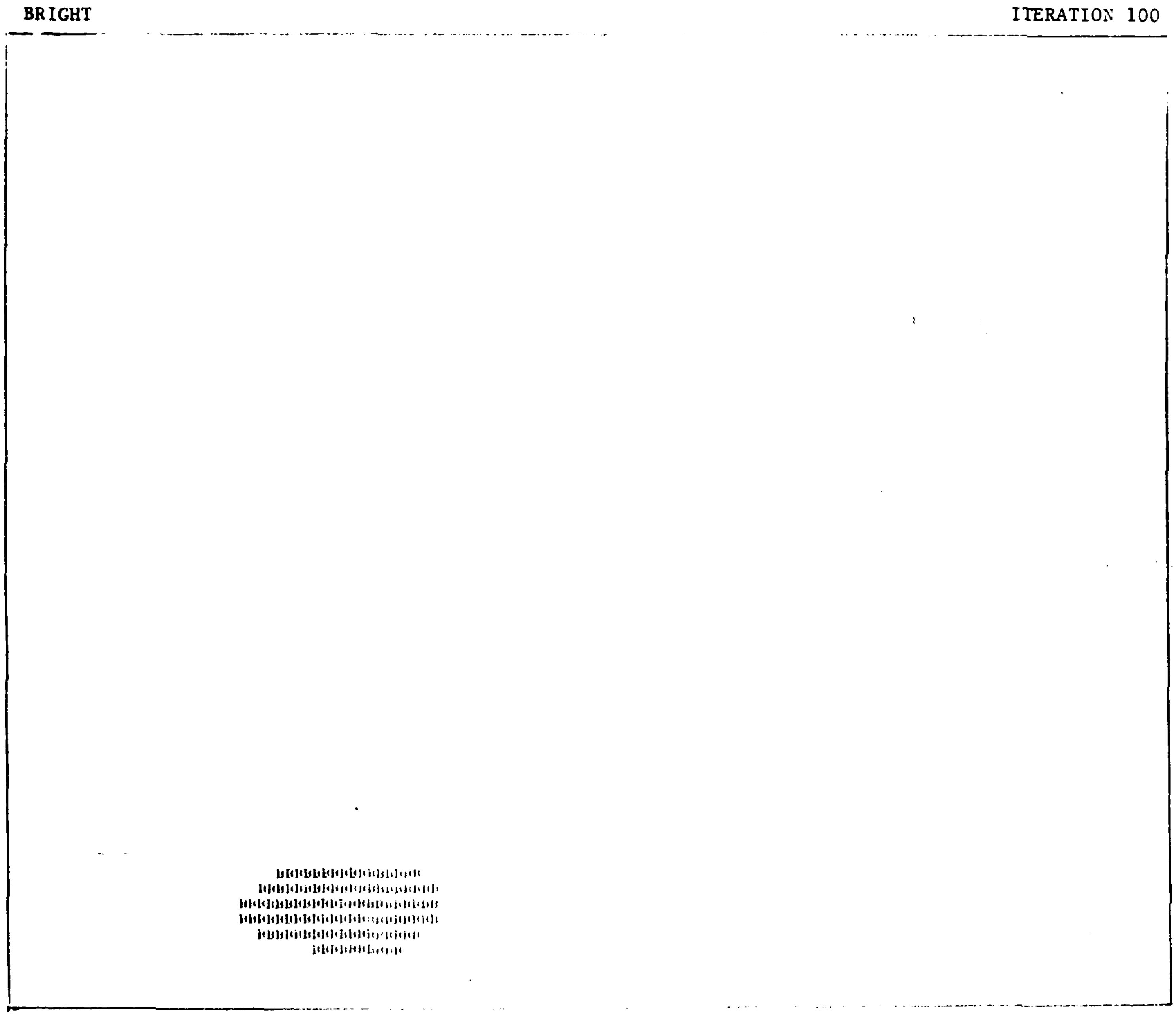


Figure 4j

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 0.10KBC= 9.00
KD= 0.00KCD= 0.00
KC= 0.10KBCD= 9.00
DELTA= 0.00
FB=G  .DIB=0,BIB=1.00
```

Figure 5

BRIGHT

ITERATION 0

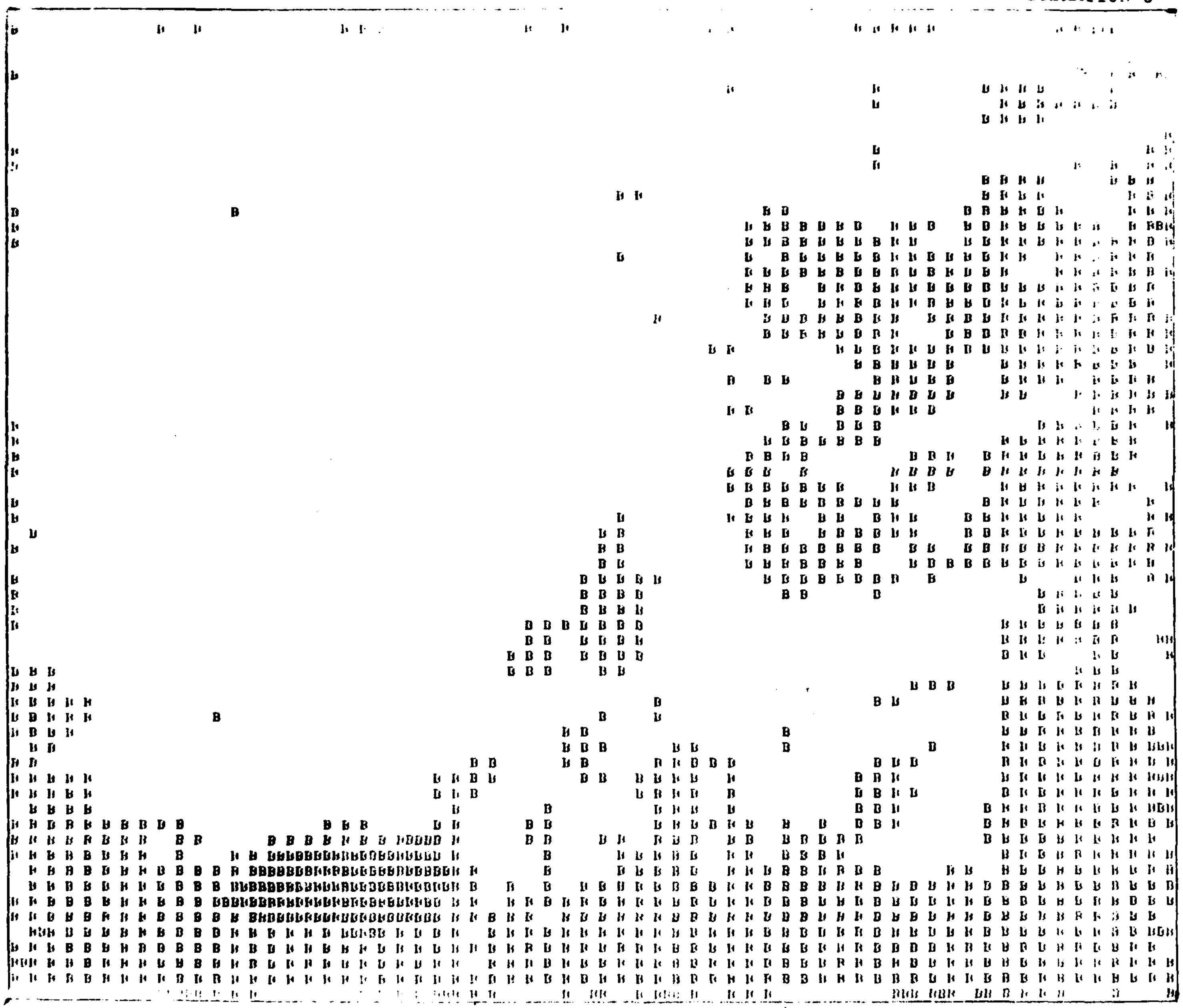


Figure 5a

Figure 5b

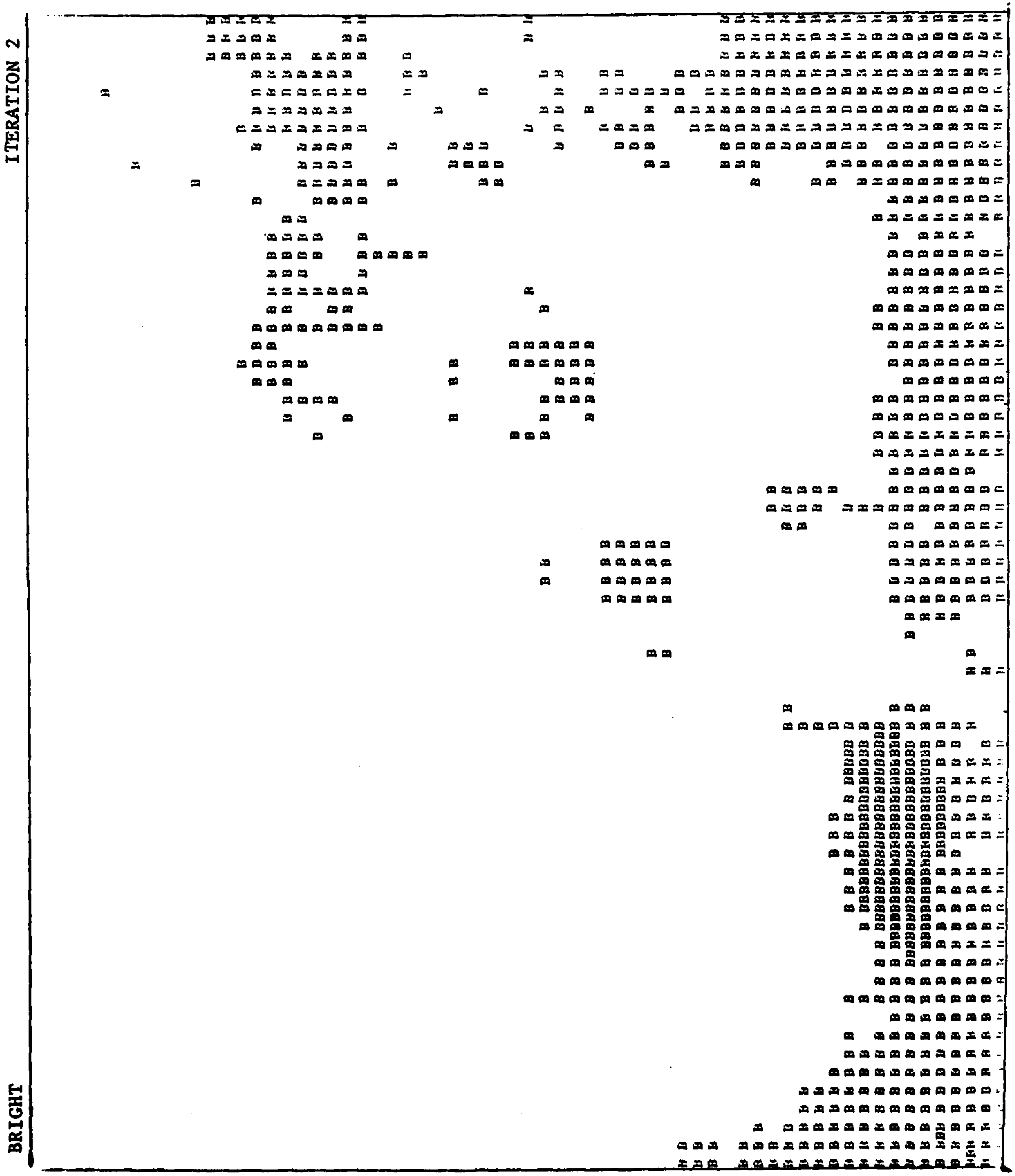


Figure 5c

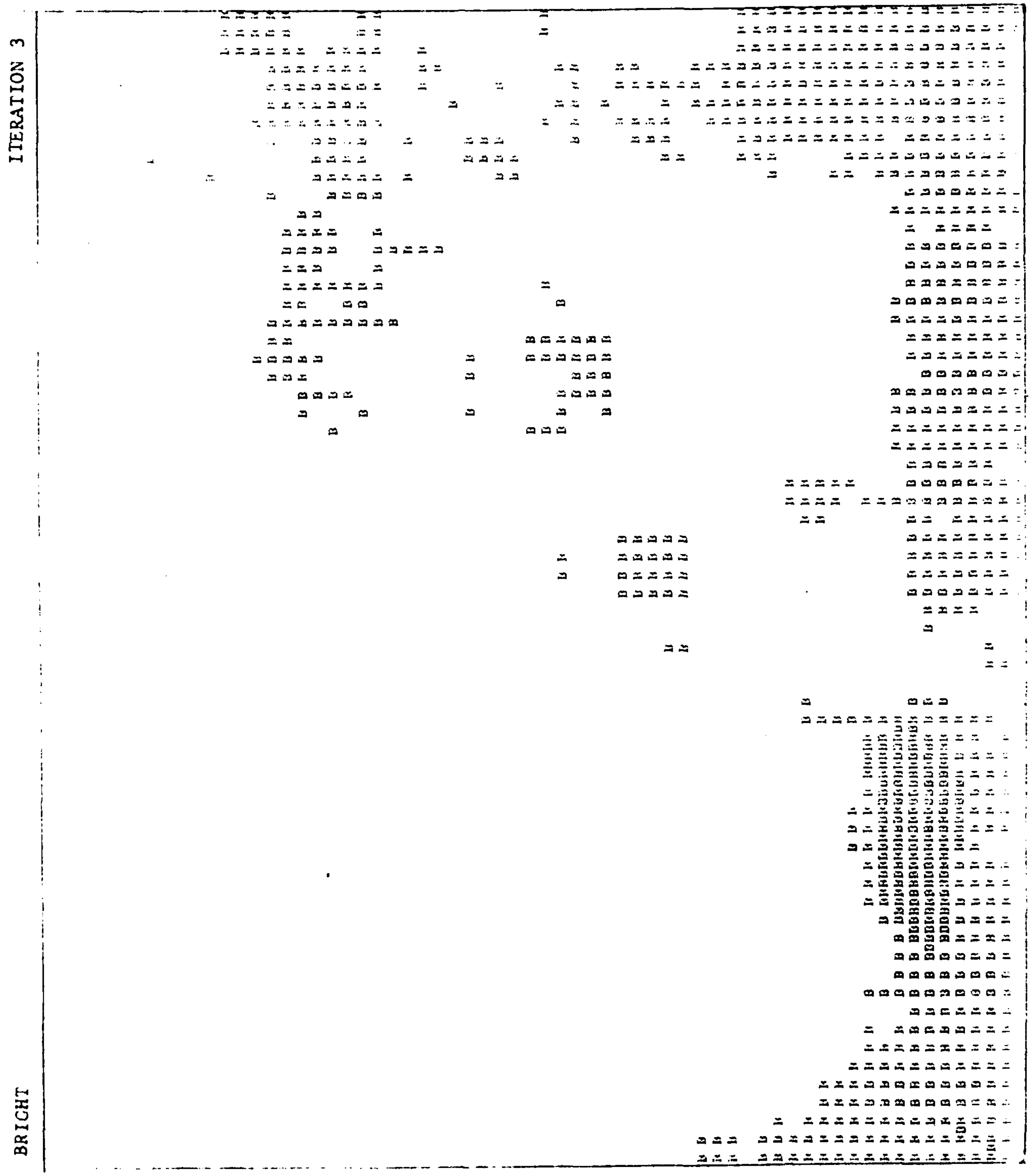


Figure 5d

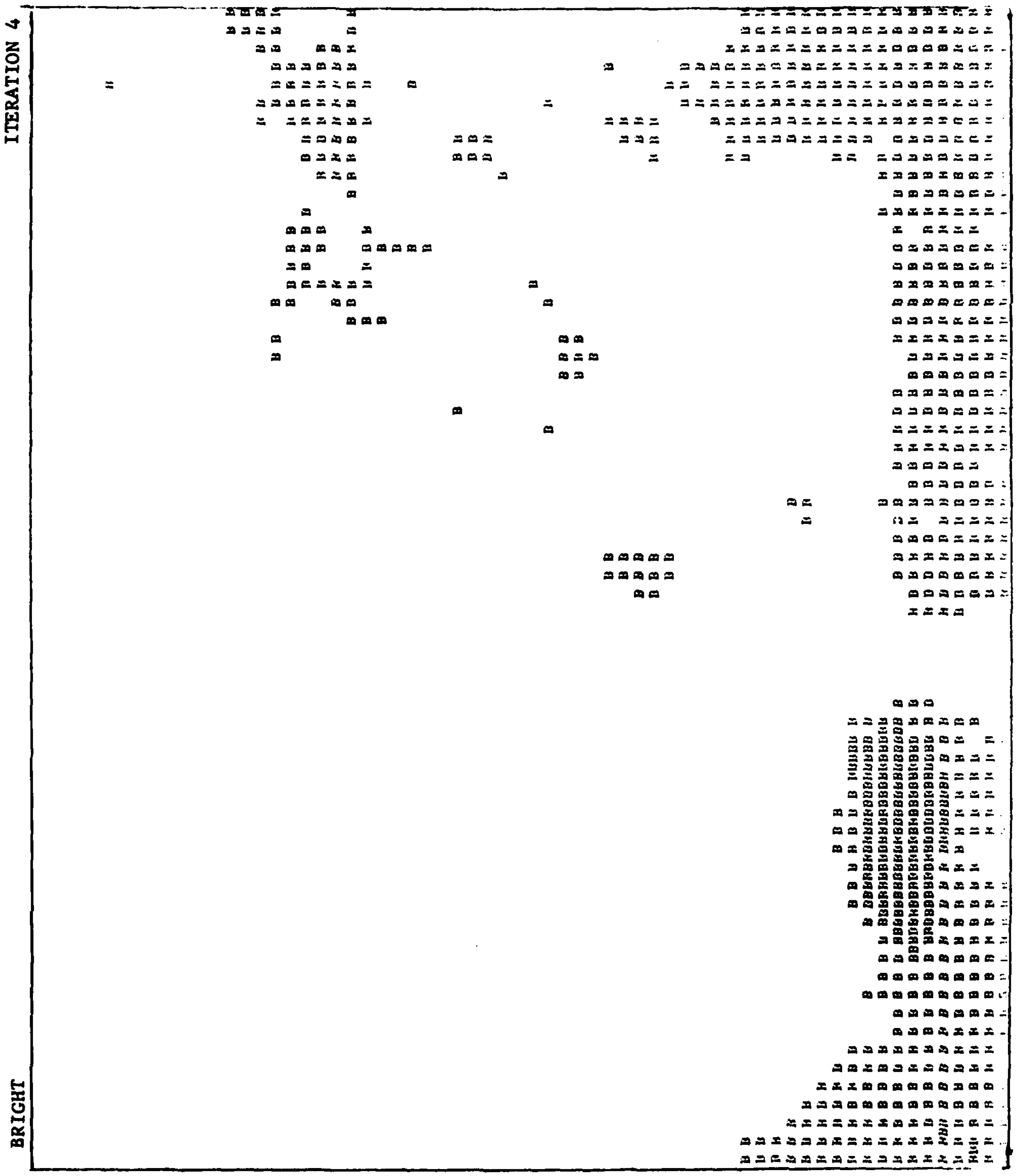


Figure 5e

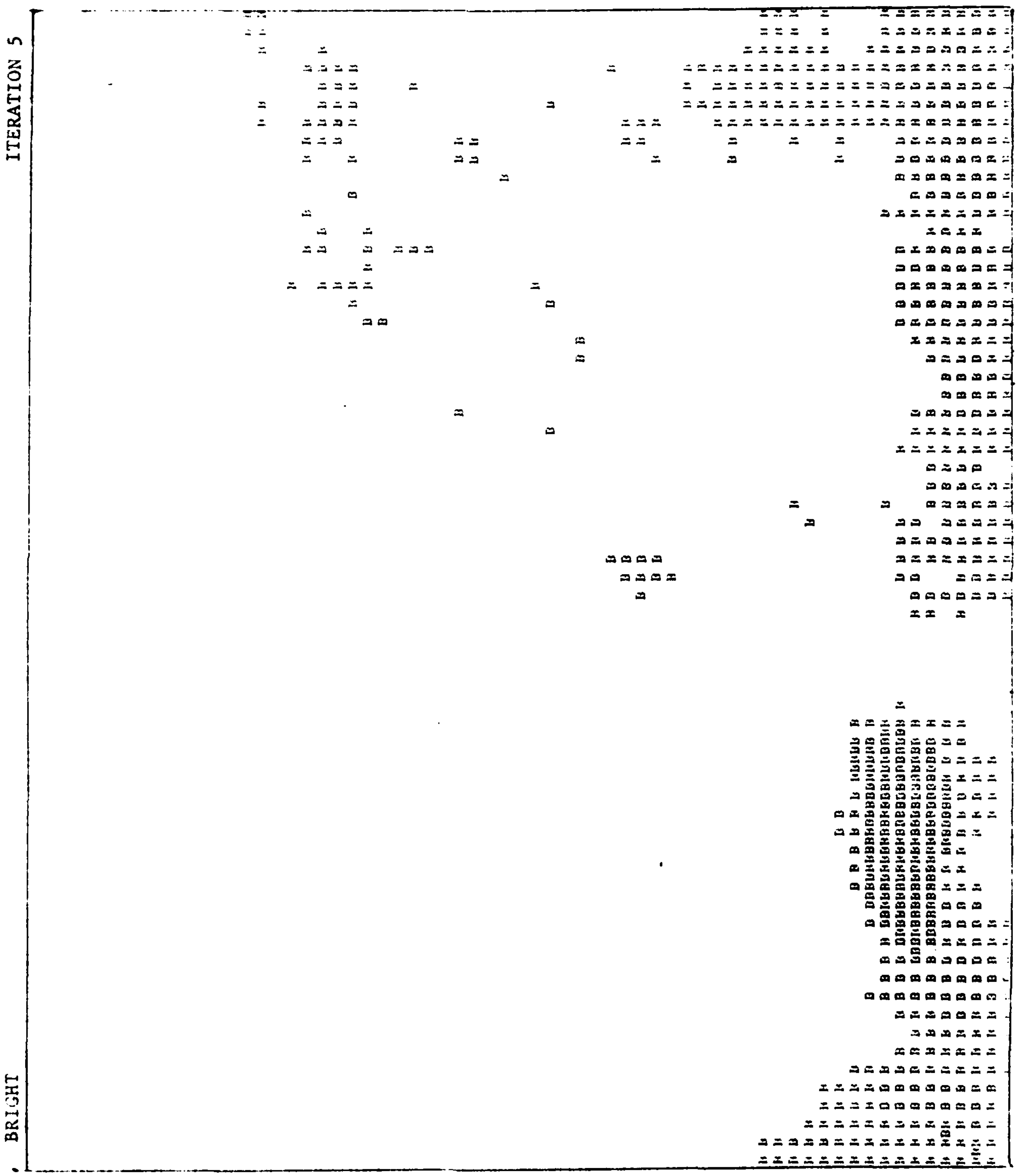


Figure 5f

BRIGHT ITERATION 6

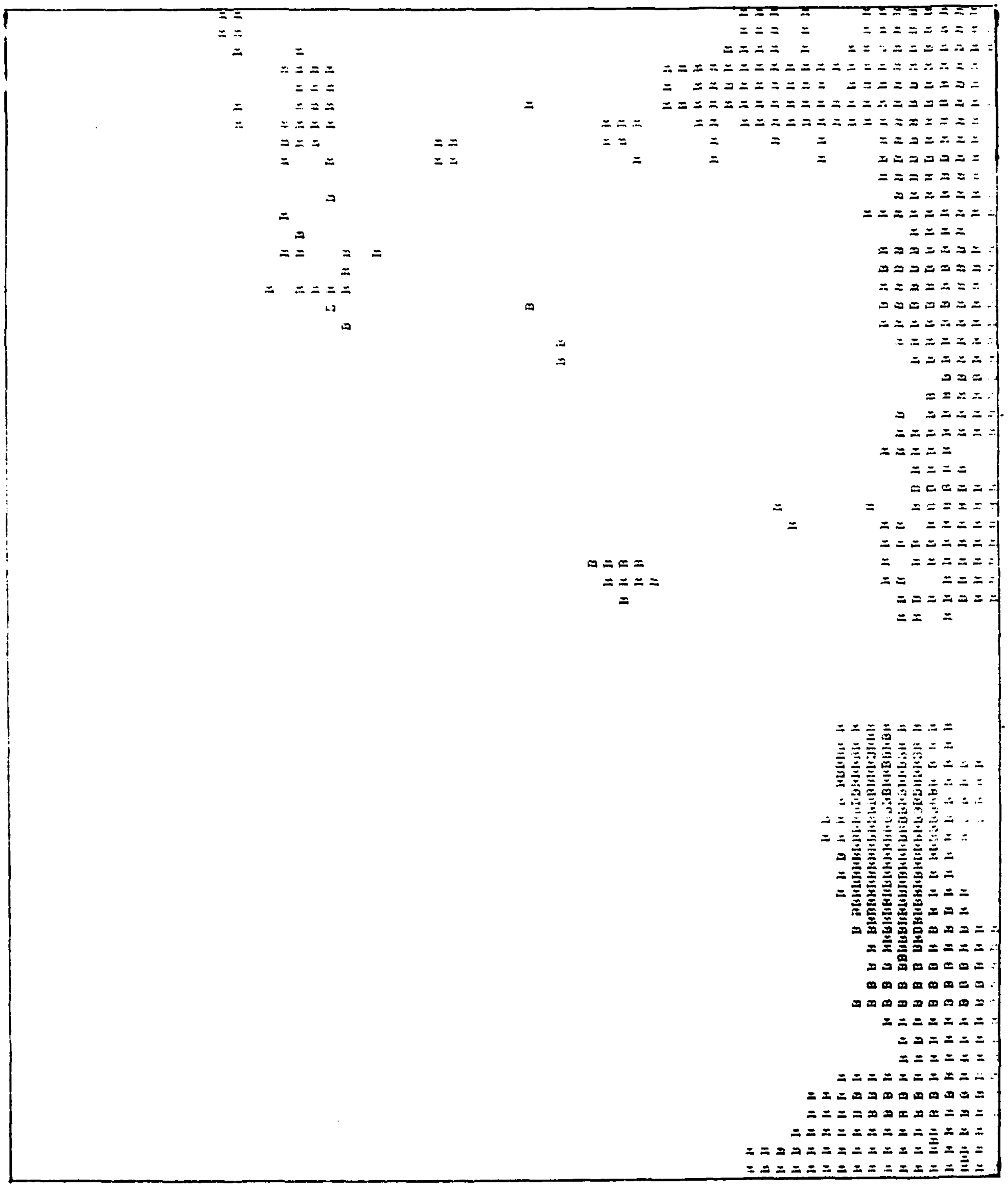


Figure 5g

BRIGHT

ITERATION 7

Figure 5h

BRIGHT

ITERATION 8

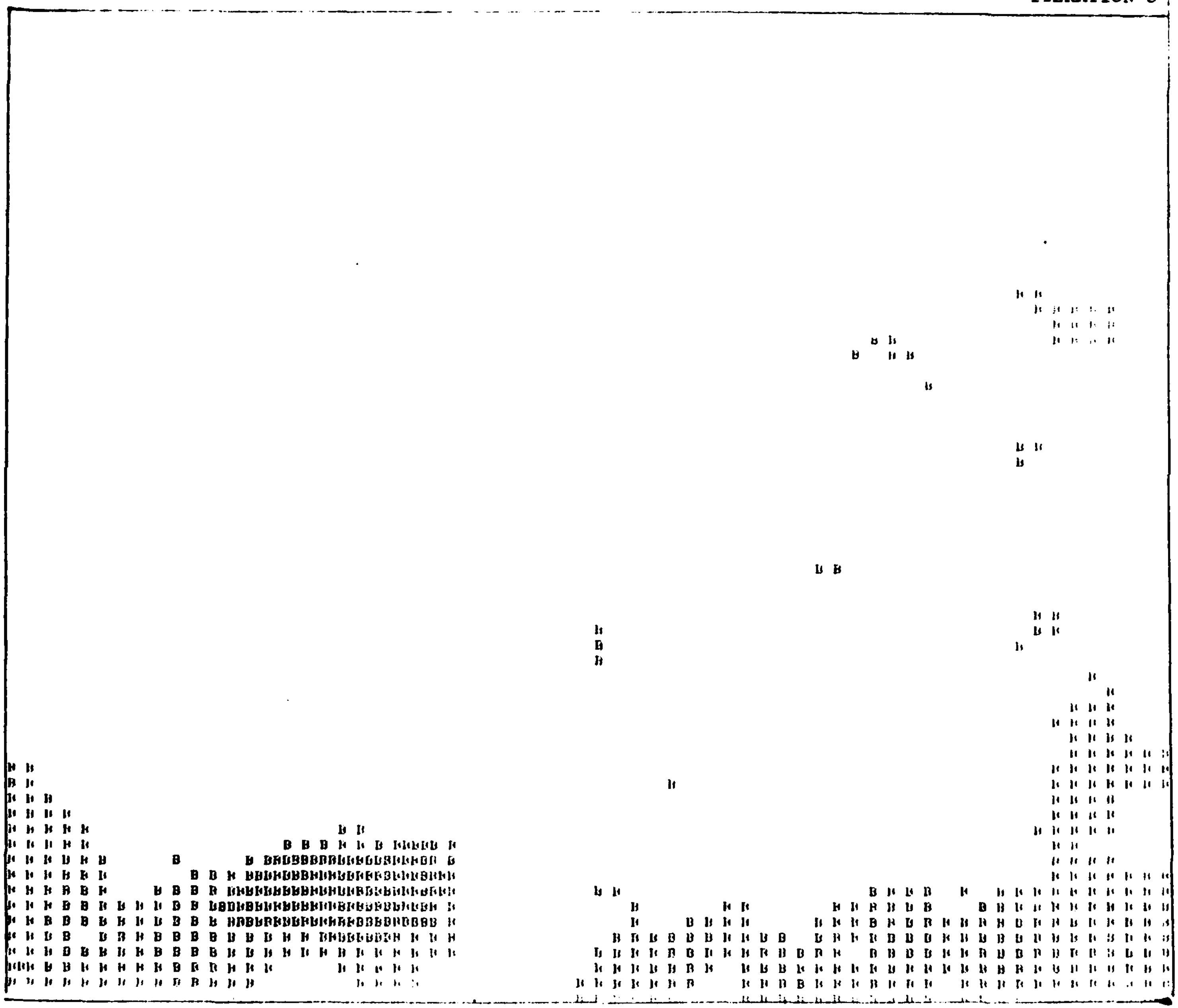


Figure 5i

BRIGHT ITERATION 10C

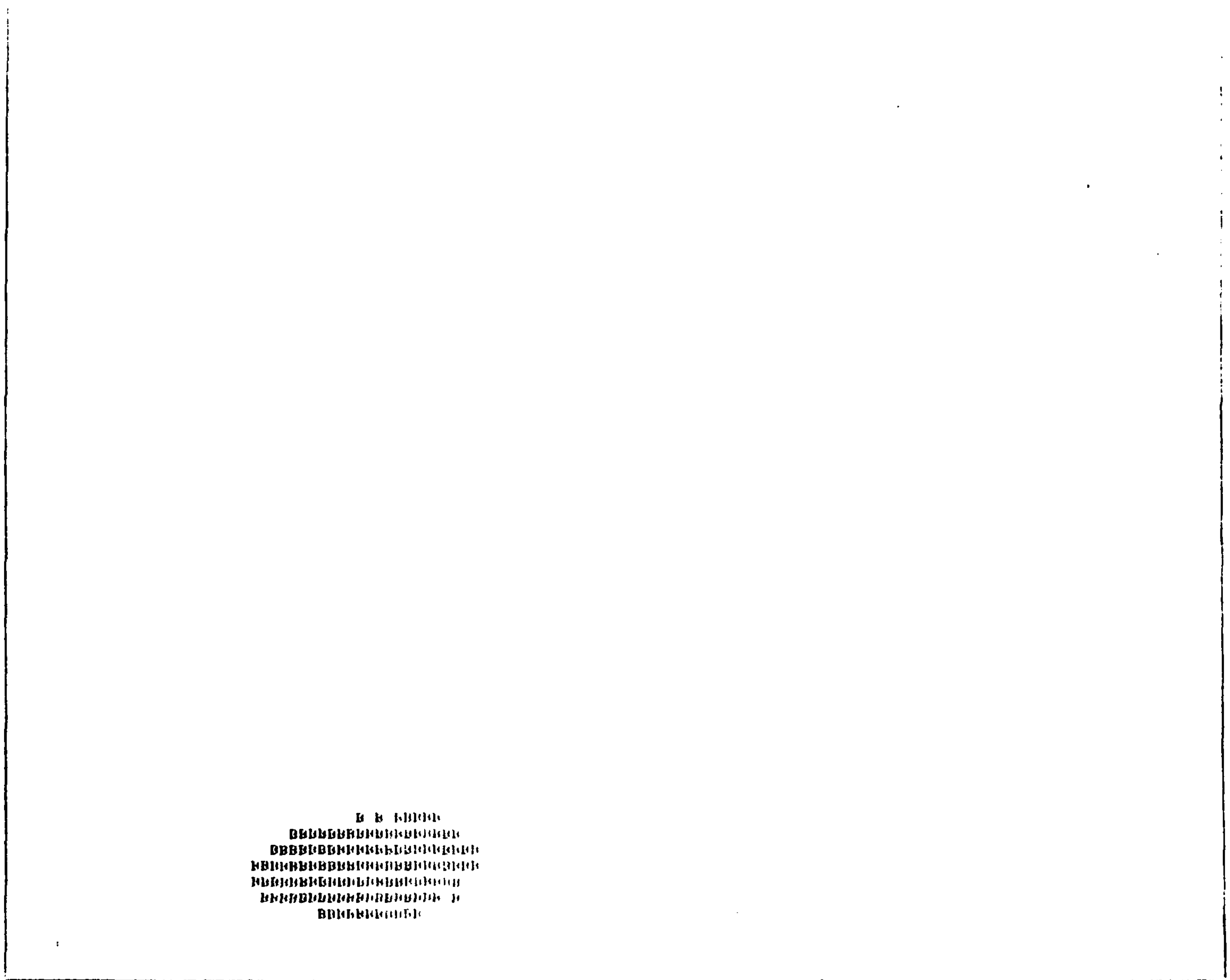


Figure 5j

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.07KBC= 0.78
KD= 0.00KCD= 0.00
KC= 0.20KBCD= 4.15
DELTA= 0.00
FB=G  .DIB=0,BIB=1.00
```

Figure 6

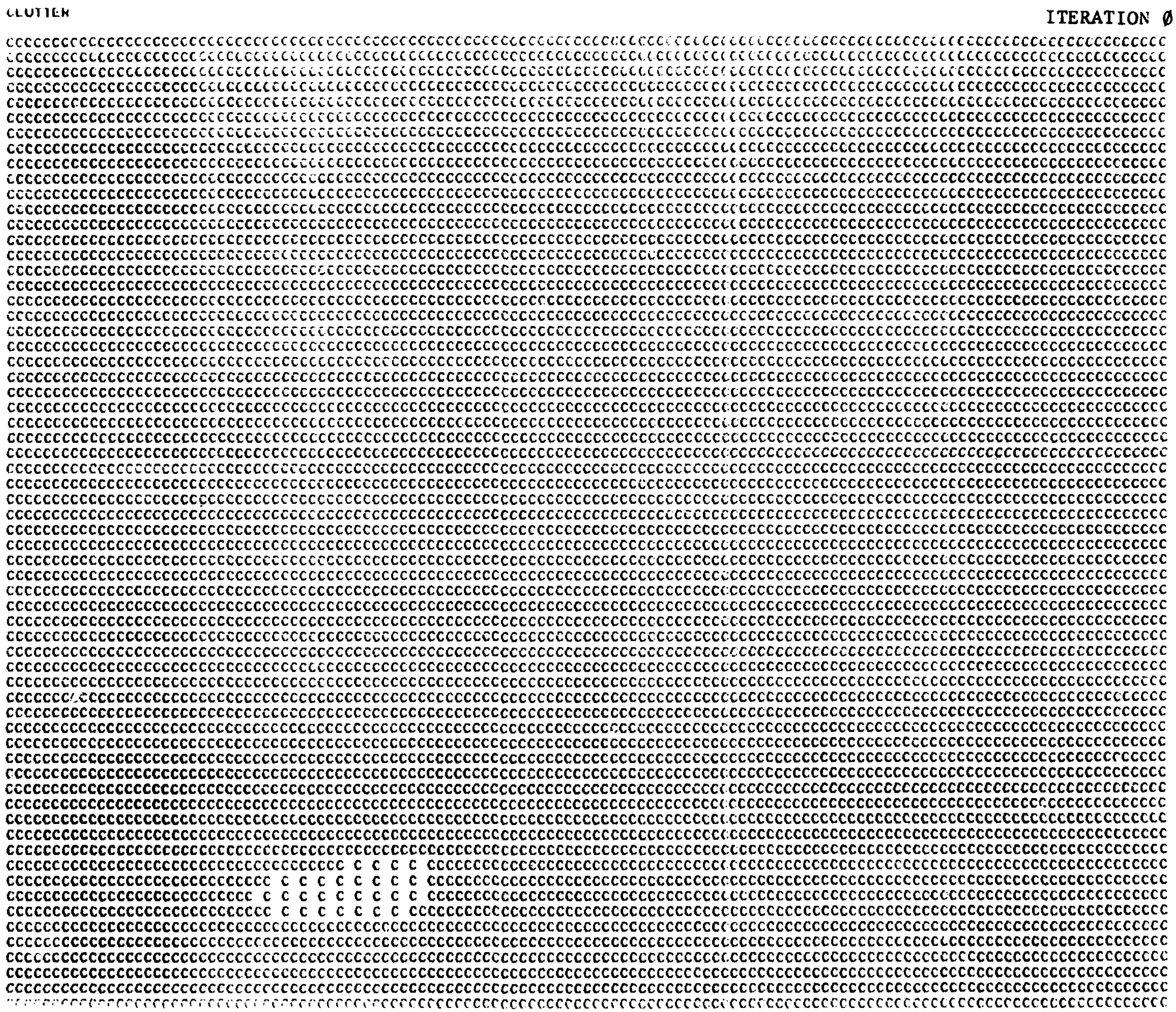


Figure 6a

CLUTTER ITERATION 1

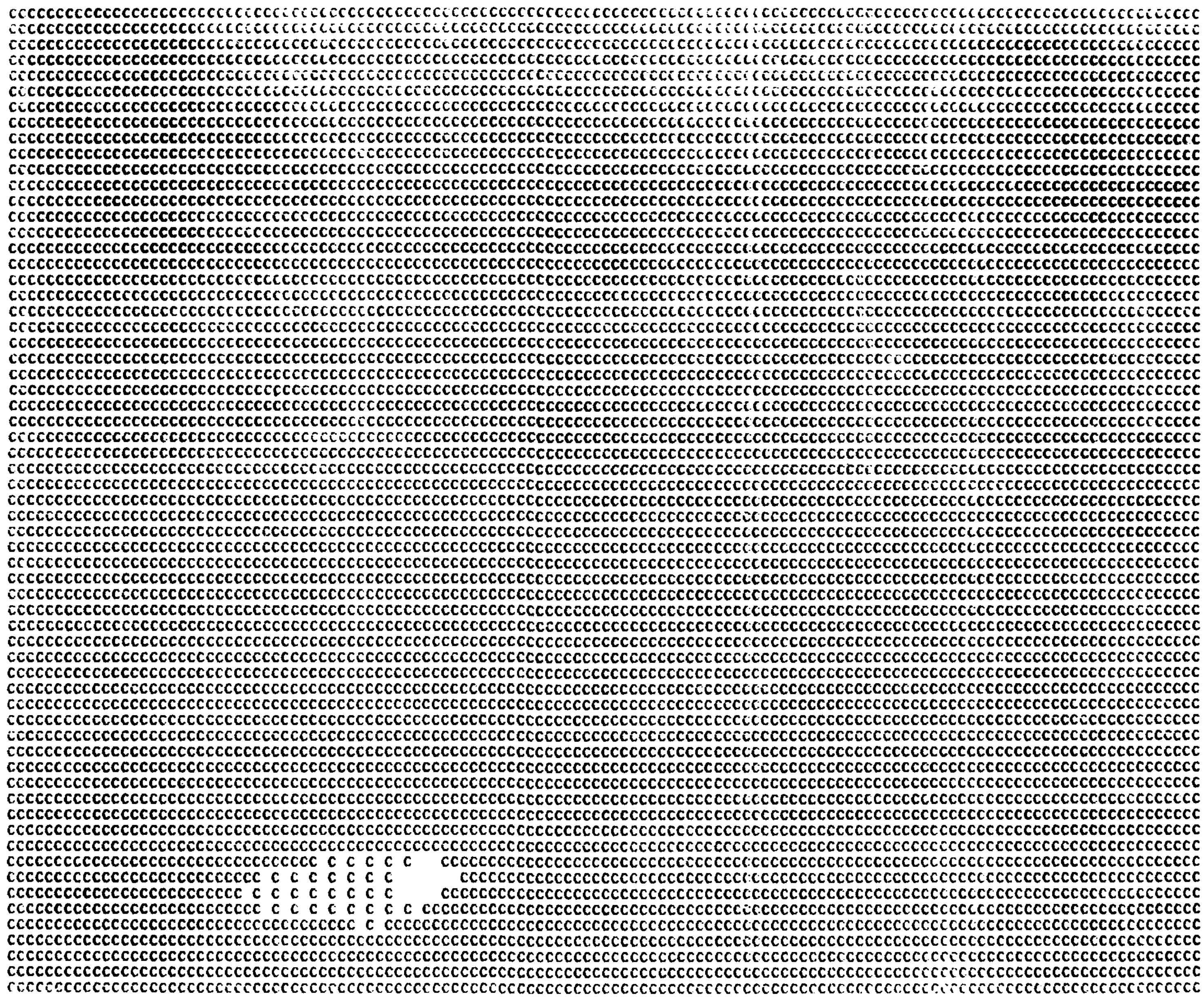


Figure 6b

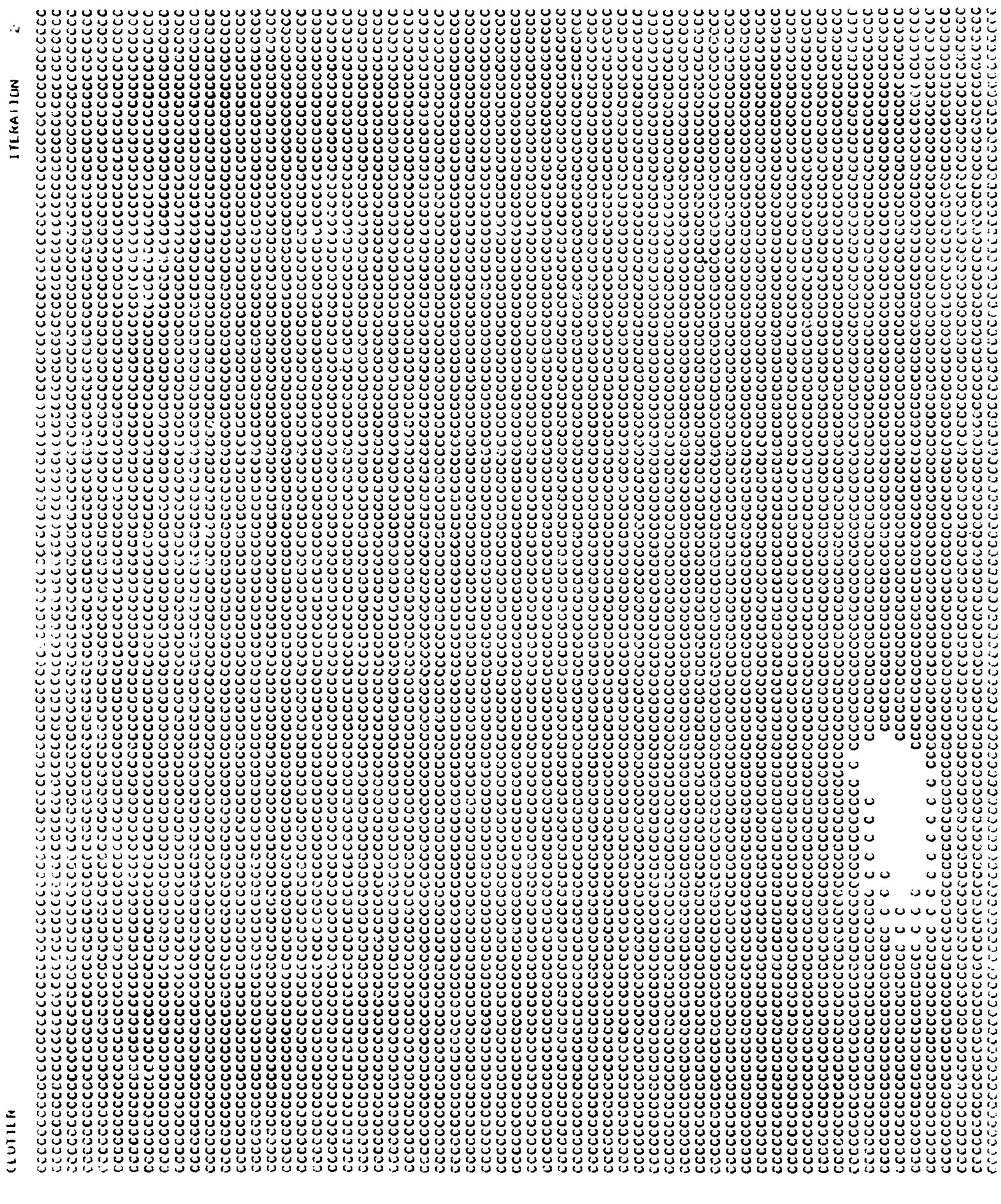


Figure 6c

PLOTTER ITERATION 3

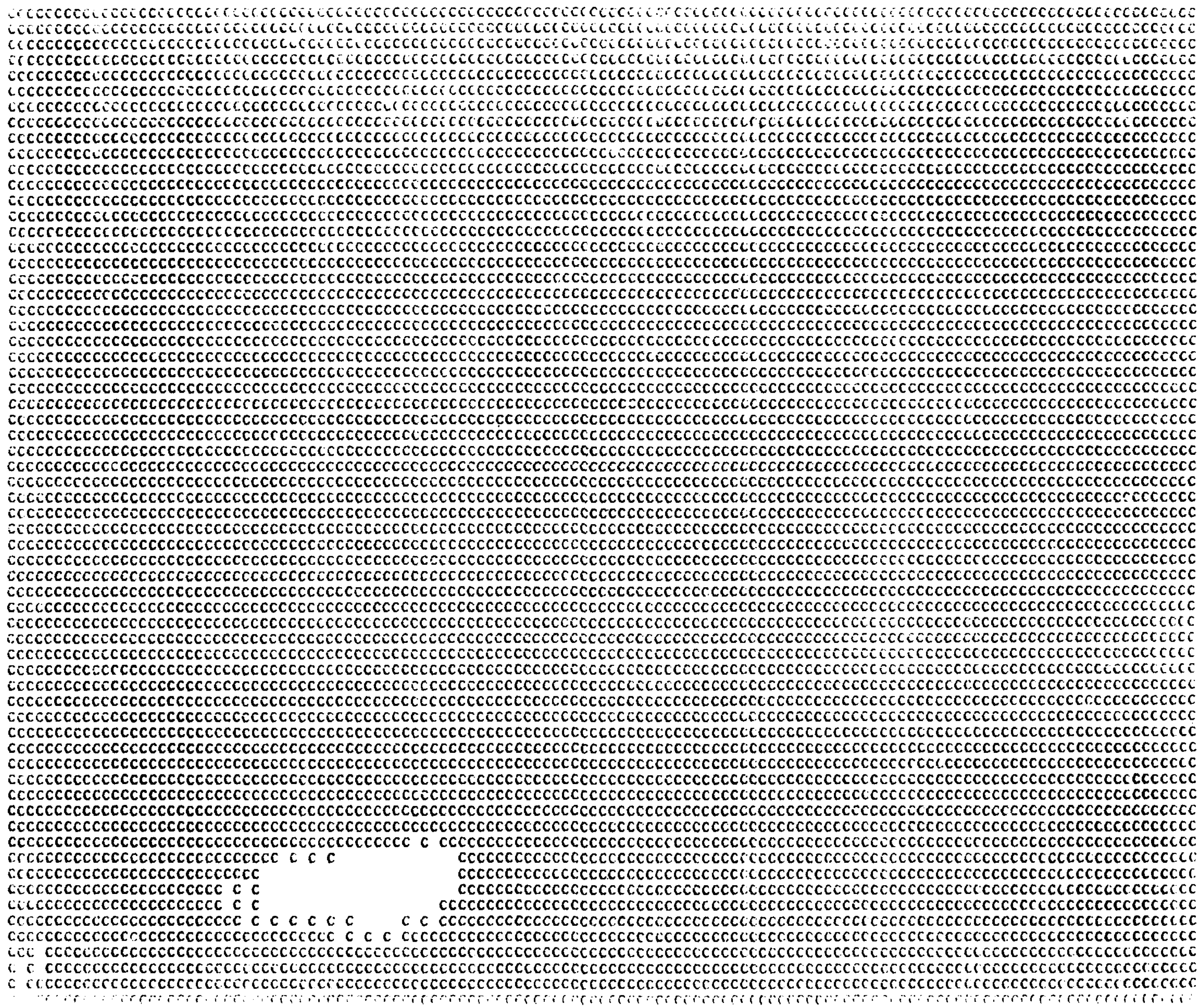


Figure 6d

Figure 6e

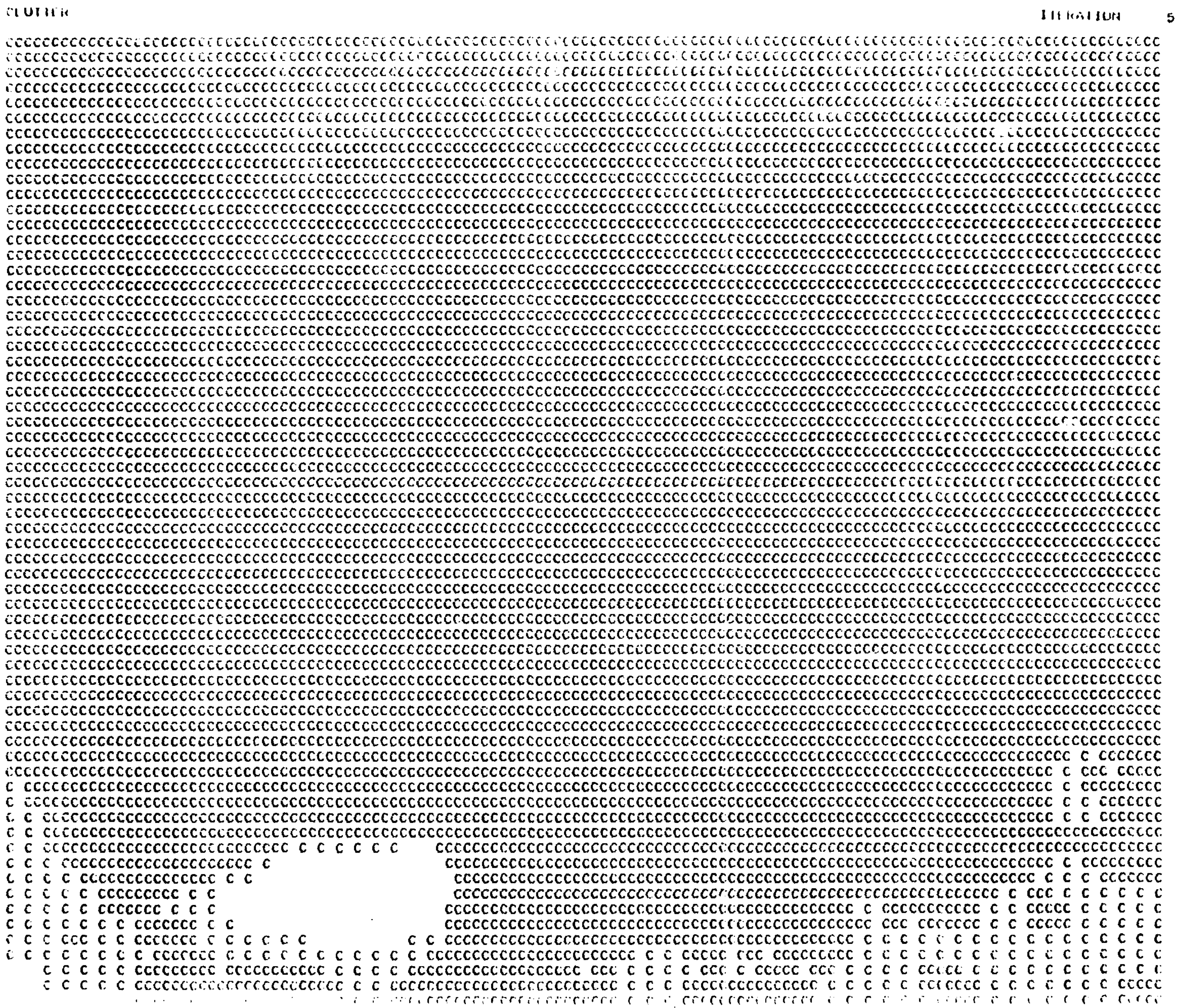


Figure 6f

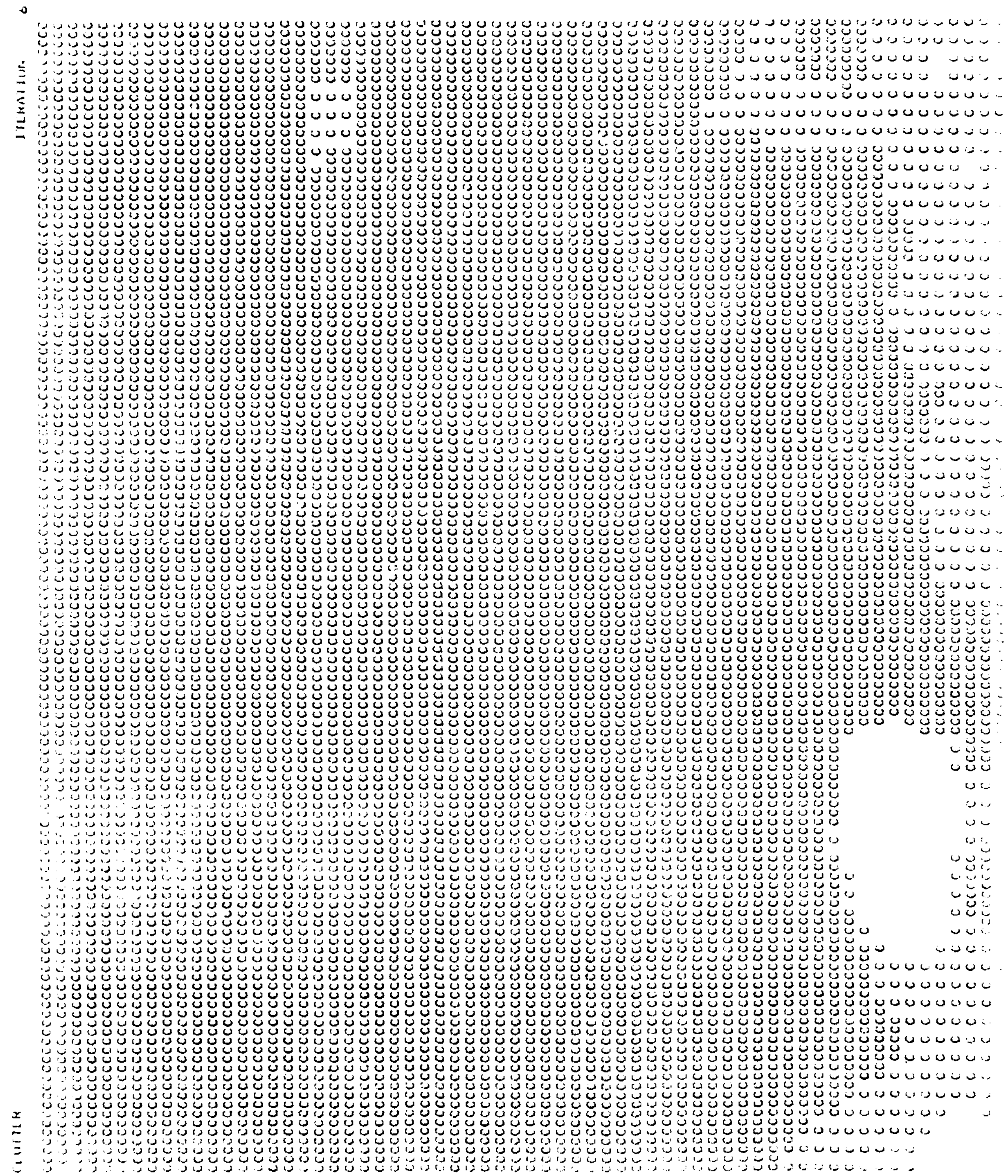


Figure 6g

CLUTTER ITERATION 7

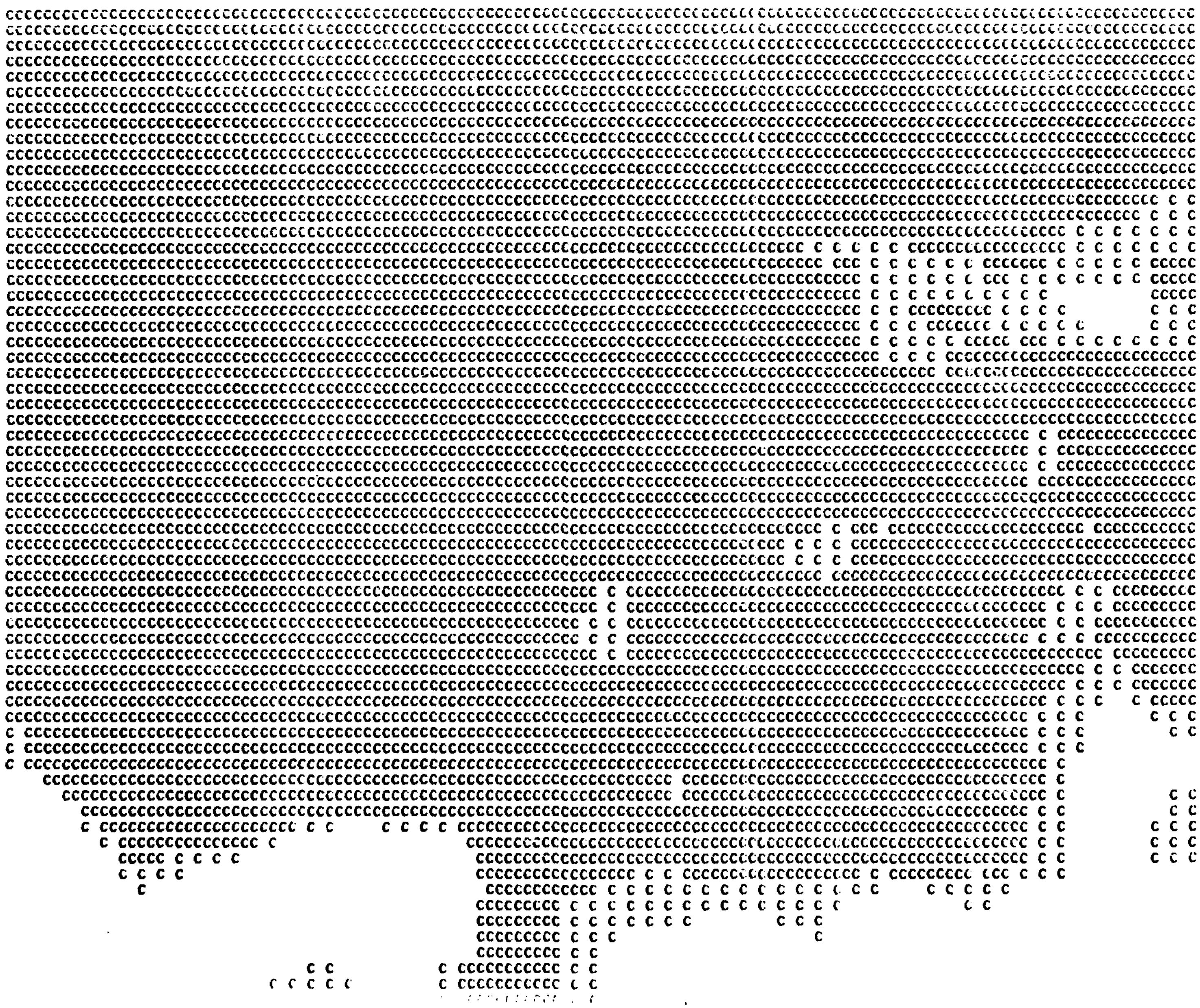


Figure 6h

Figure 6i

PLOTTER ITERATION 100

Figure 6j

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.07KBC= 0.78
KD= 0.00KCD= 0.00
KC= 0.20KBCD= 4.15
DELTA= 0.00
FB=G  .DIB=0,BIB=0.85
```

Figure 7

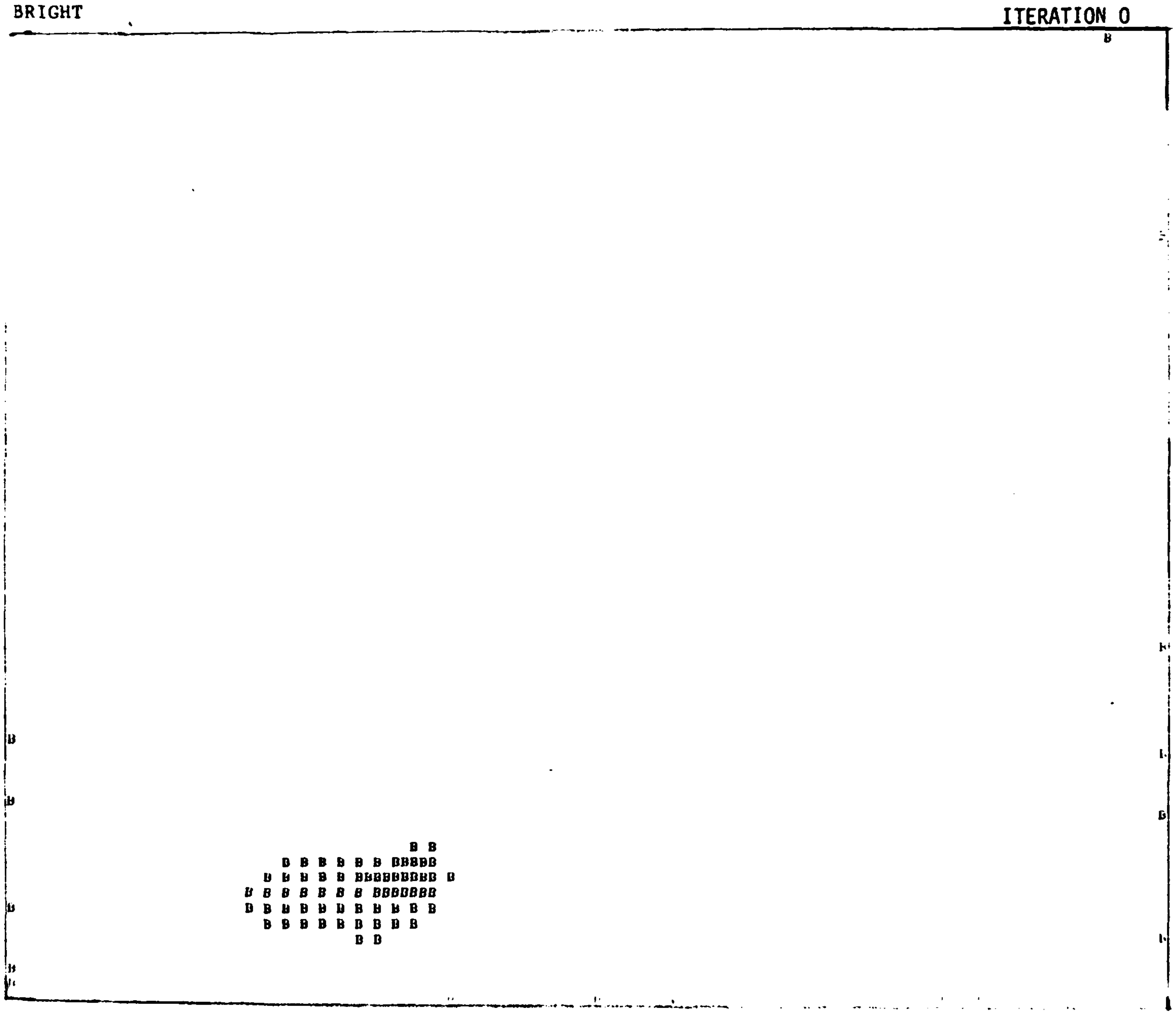


Figure 7a

BRIGHT ITERATION 1

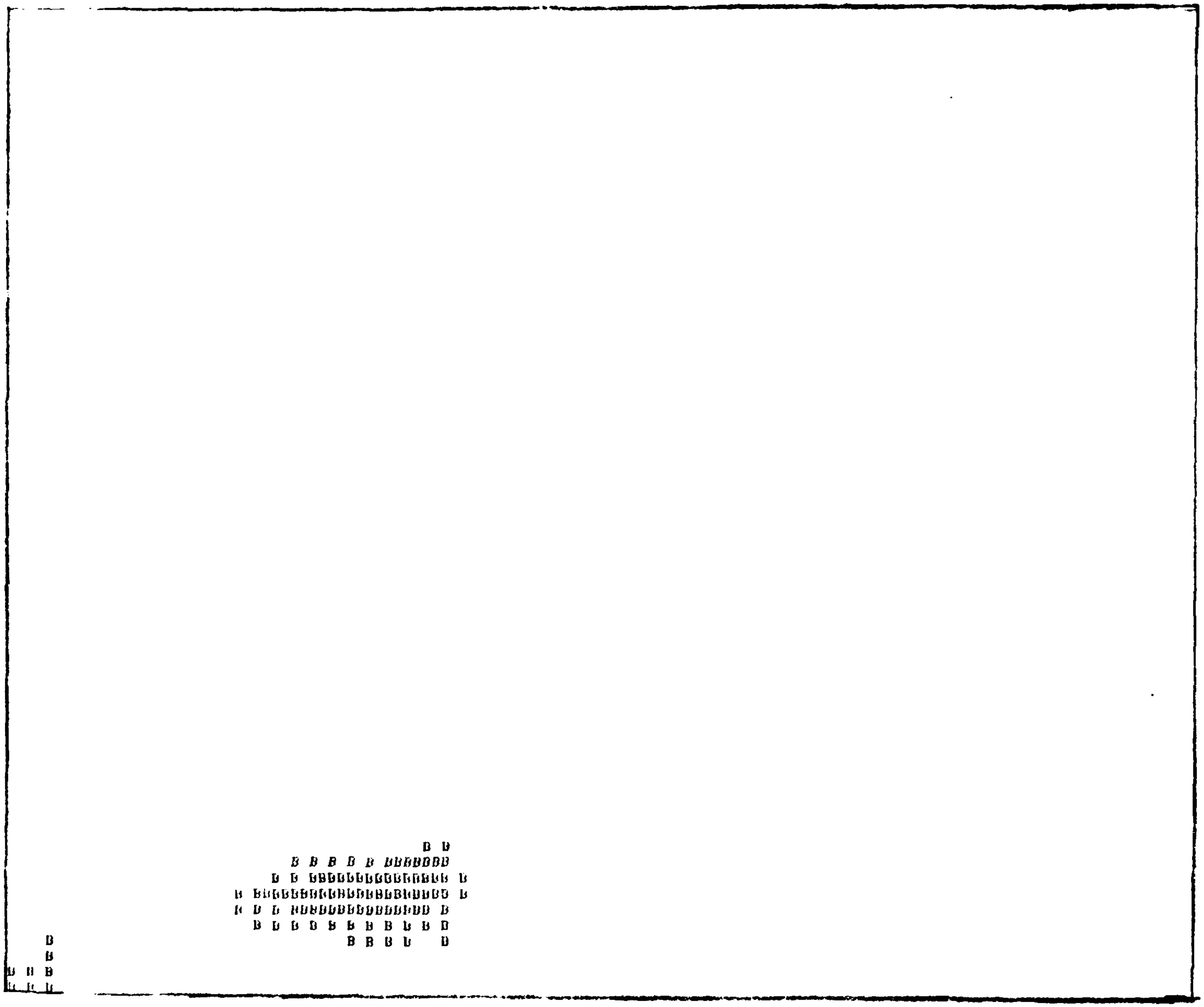


Figure 7b

BRIGHT

ITERATION 2

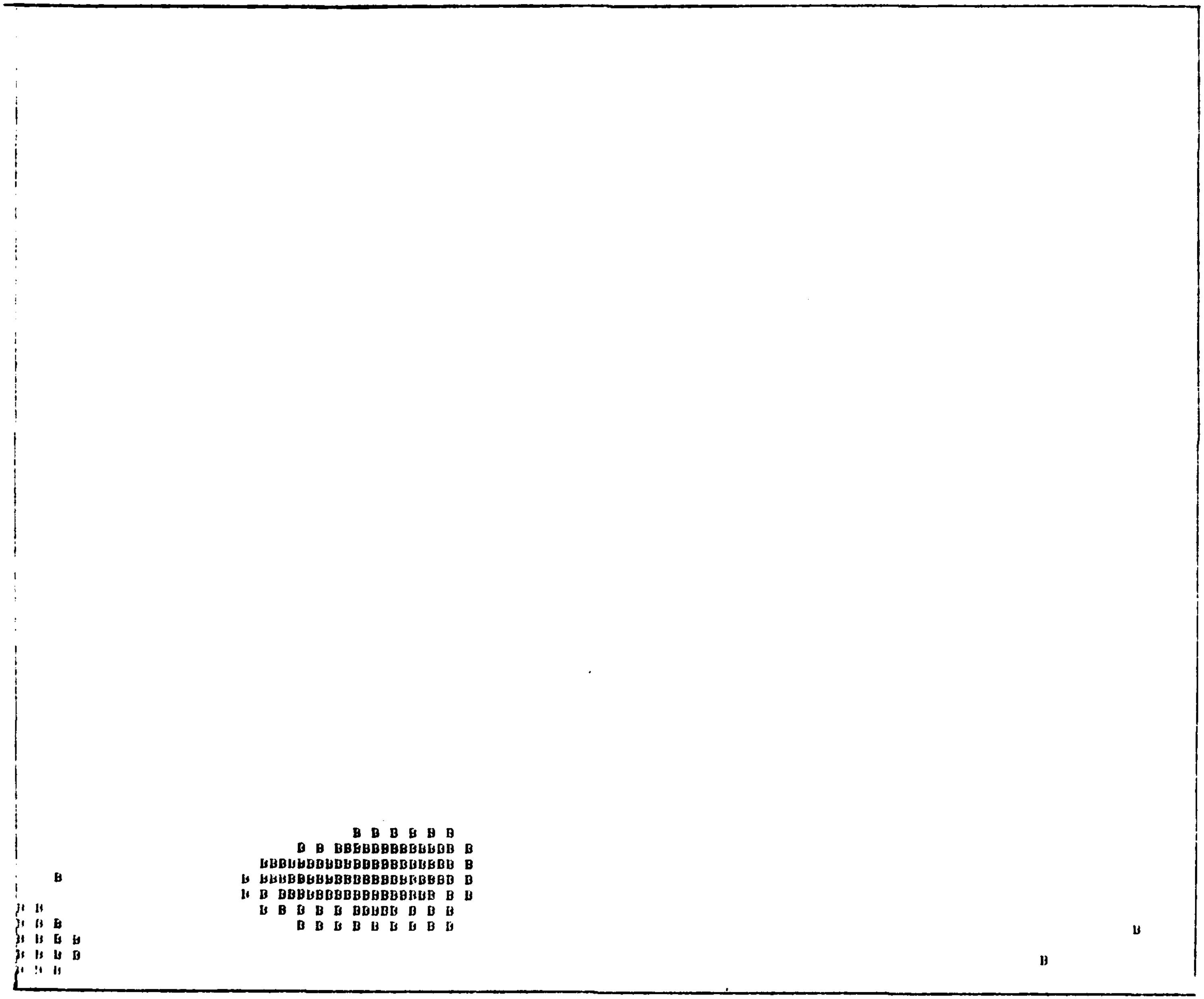


Figure 7c

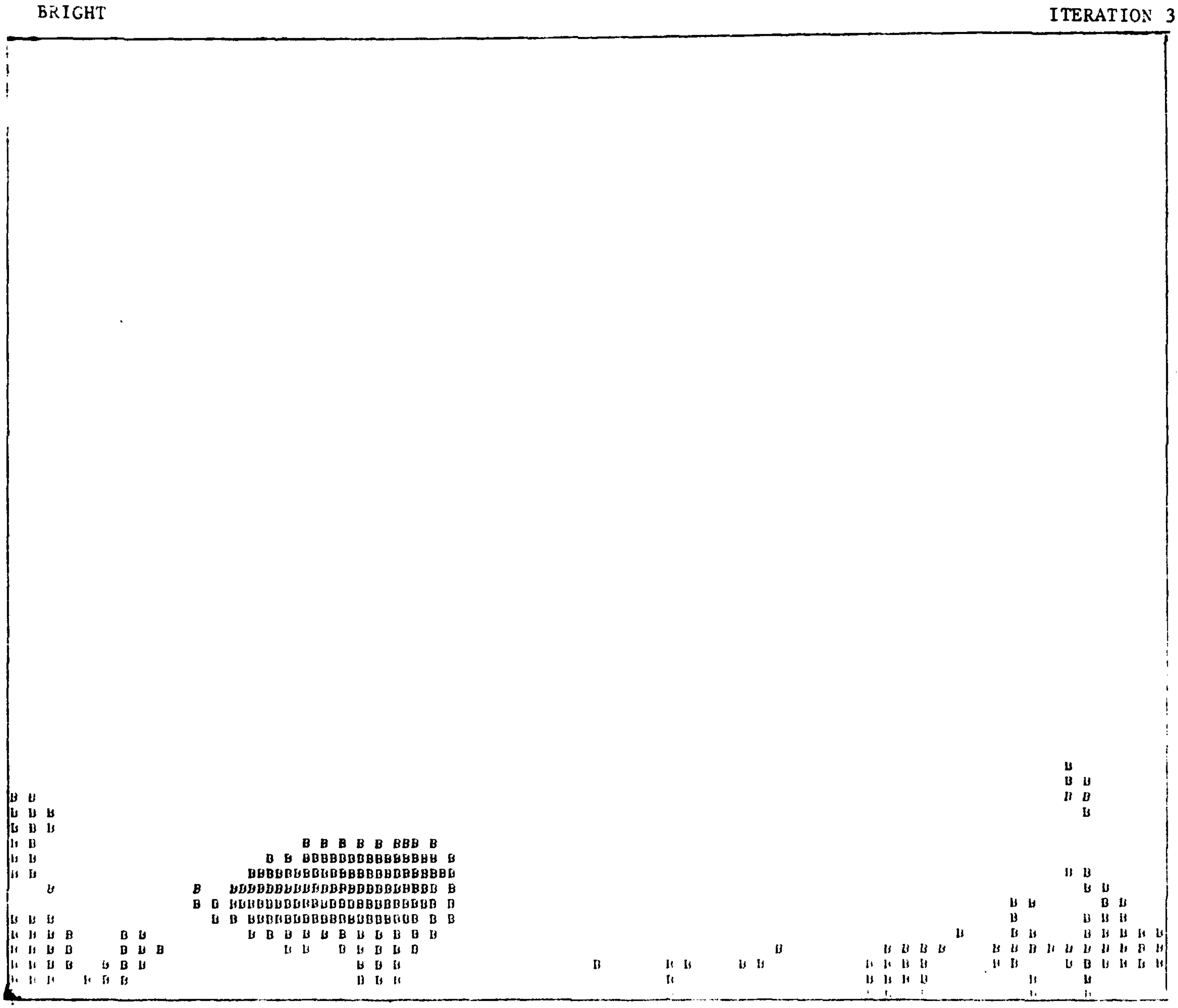


Figure 7d

BRIGHT ITERATION 4

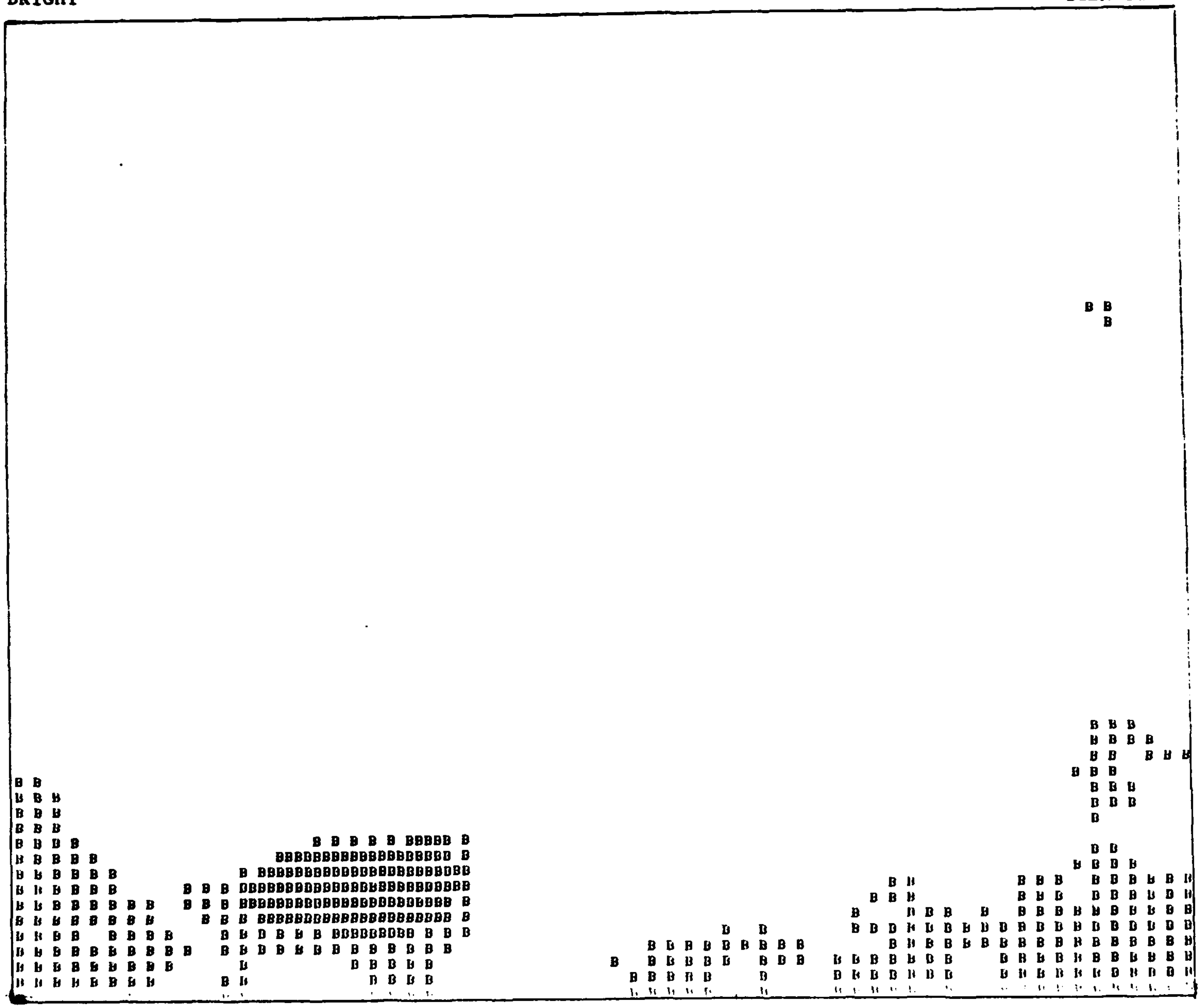


Figure 7e

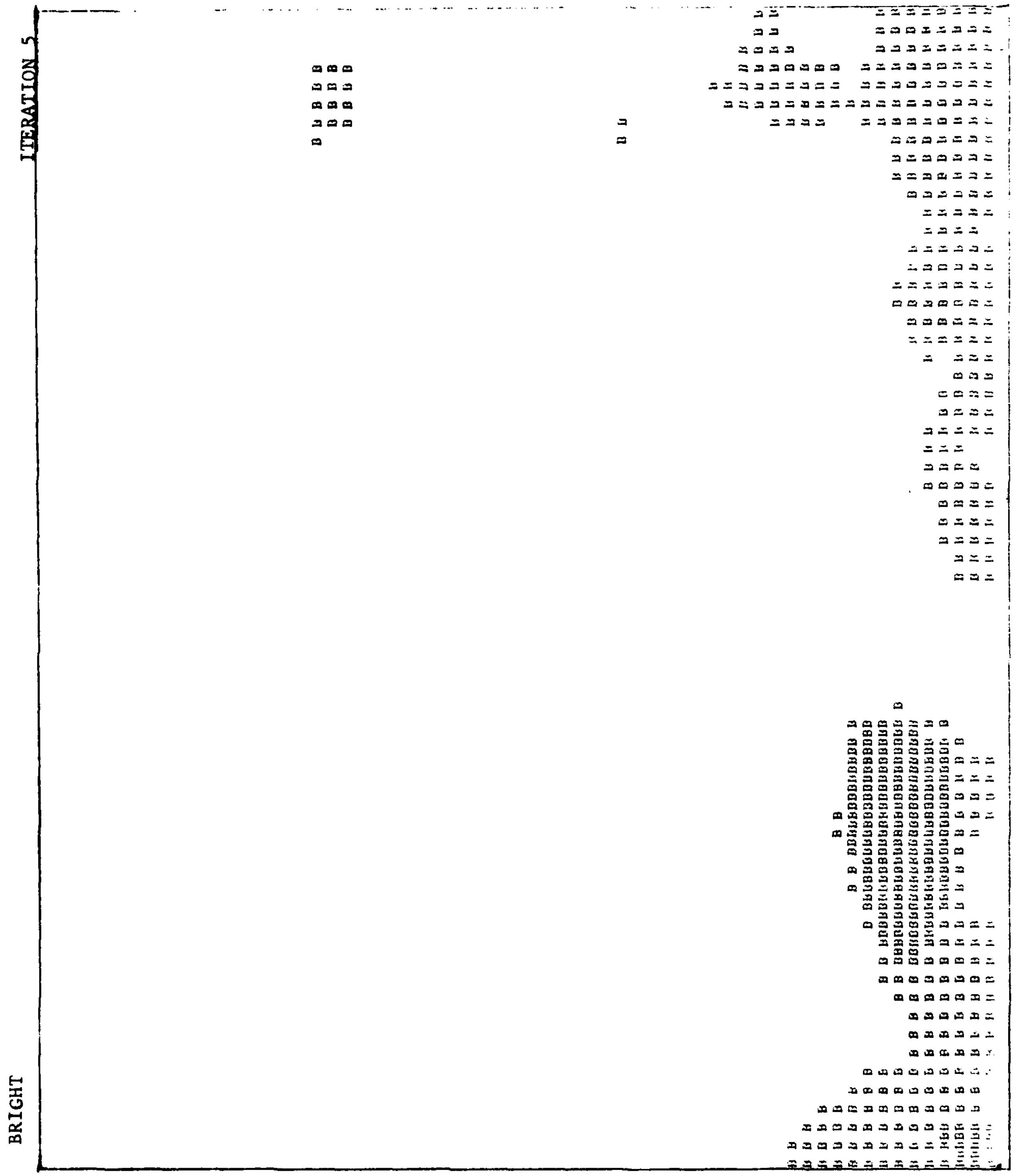


Figure 7f

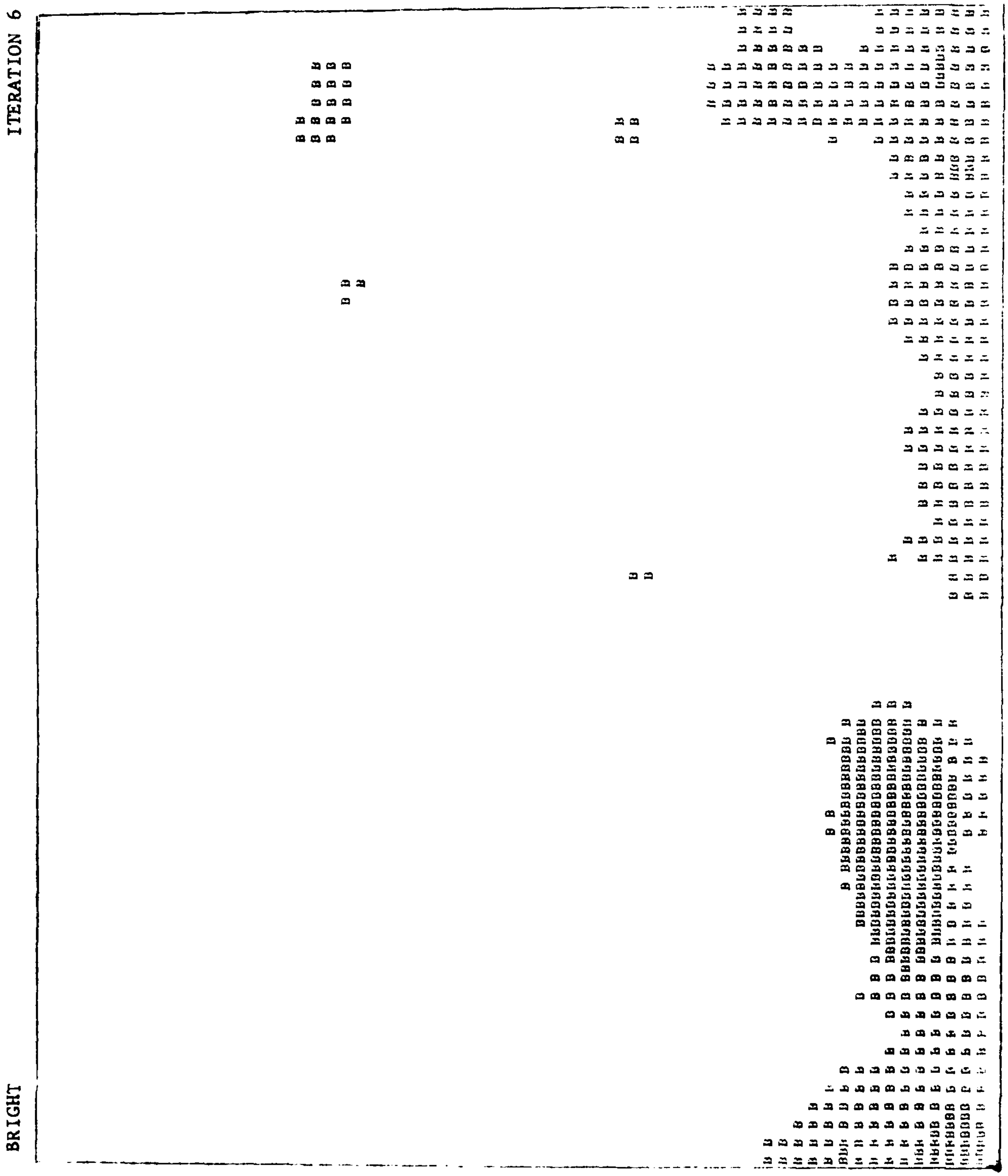


Figure 7g

BRIGHT

ITERATION 7

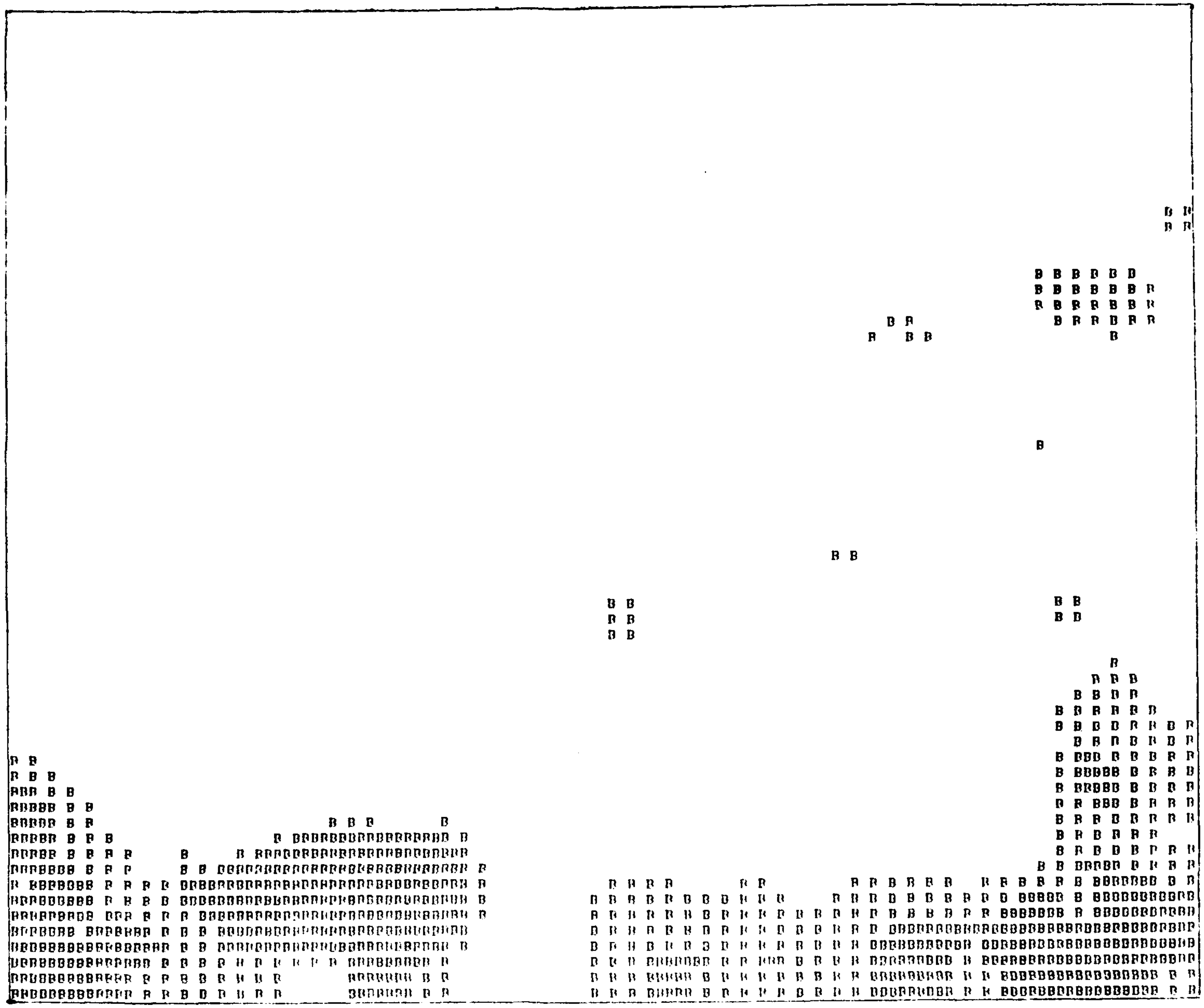


Figure 7h

BRIGHT ITERATION 8

Figure 7i

BRIGHT

ITERATION 100

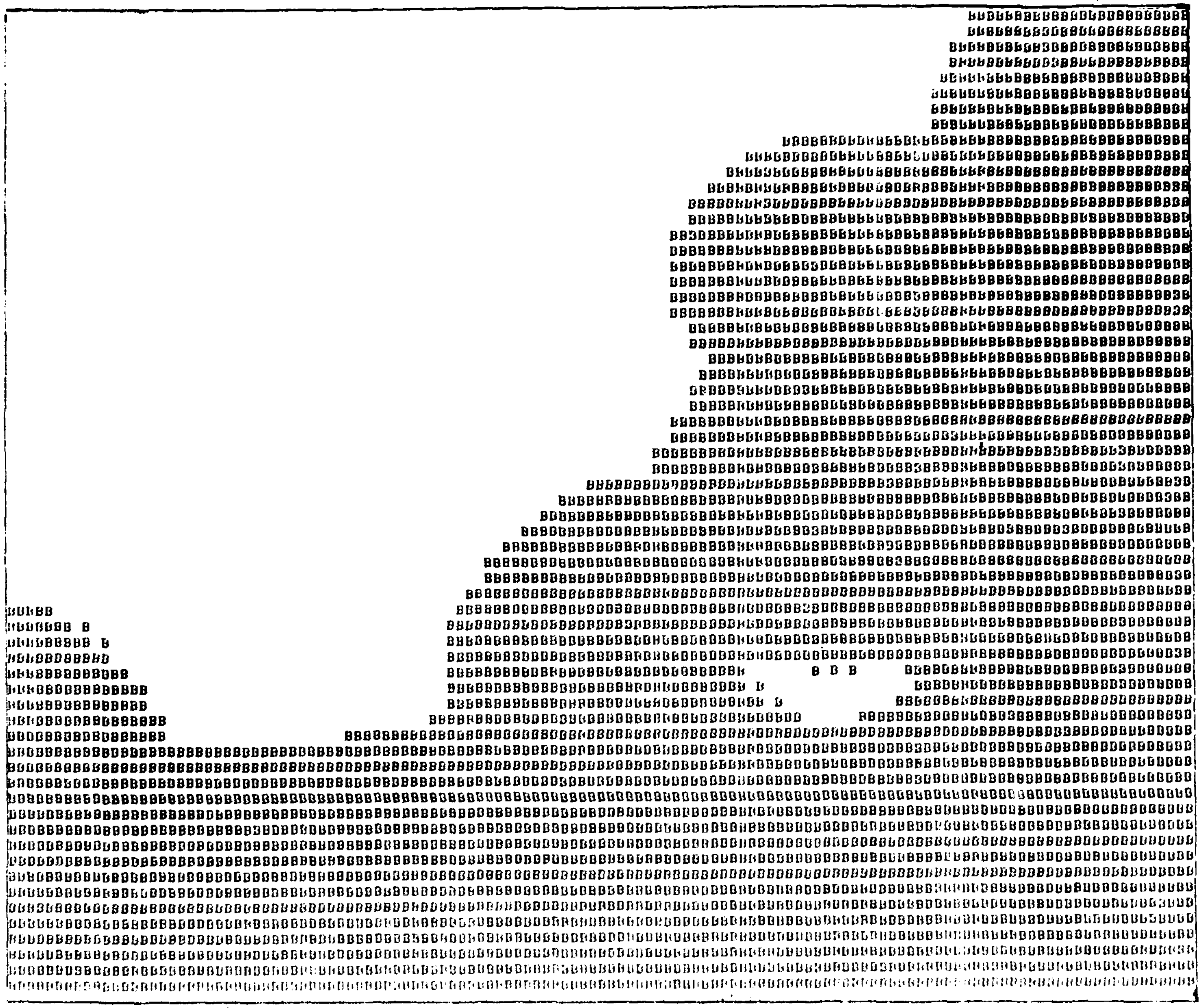


Figure 7j

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.07KBC= 0.78
KD= 0.00KCD= 0.00
KC= 0.20KBCD= 4.15
DELTA= 0.00
FB=G  .DIB=0, BIB=0.65
```

Figure 8

BRIGHT

ITERATION 0

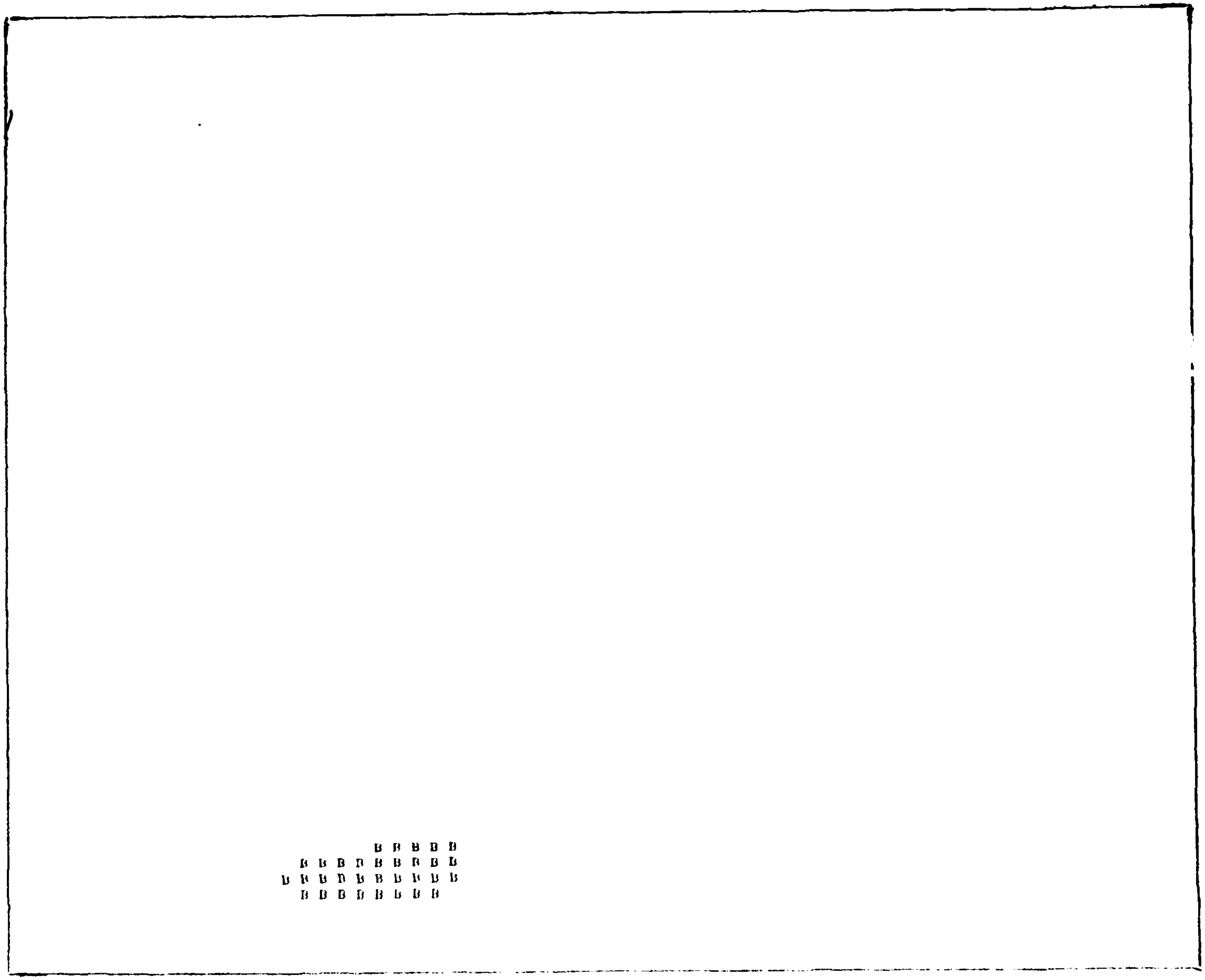


Figure 8a

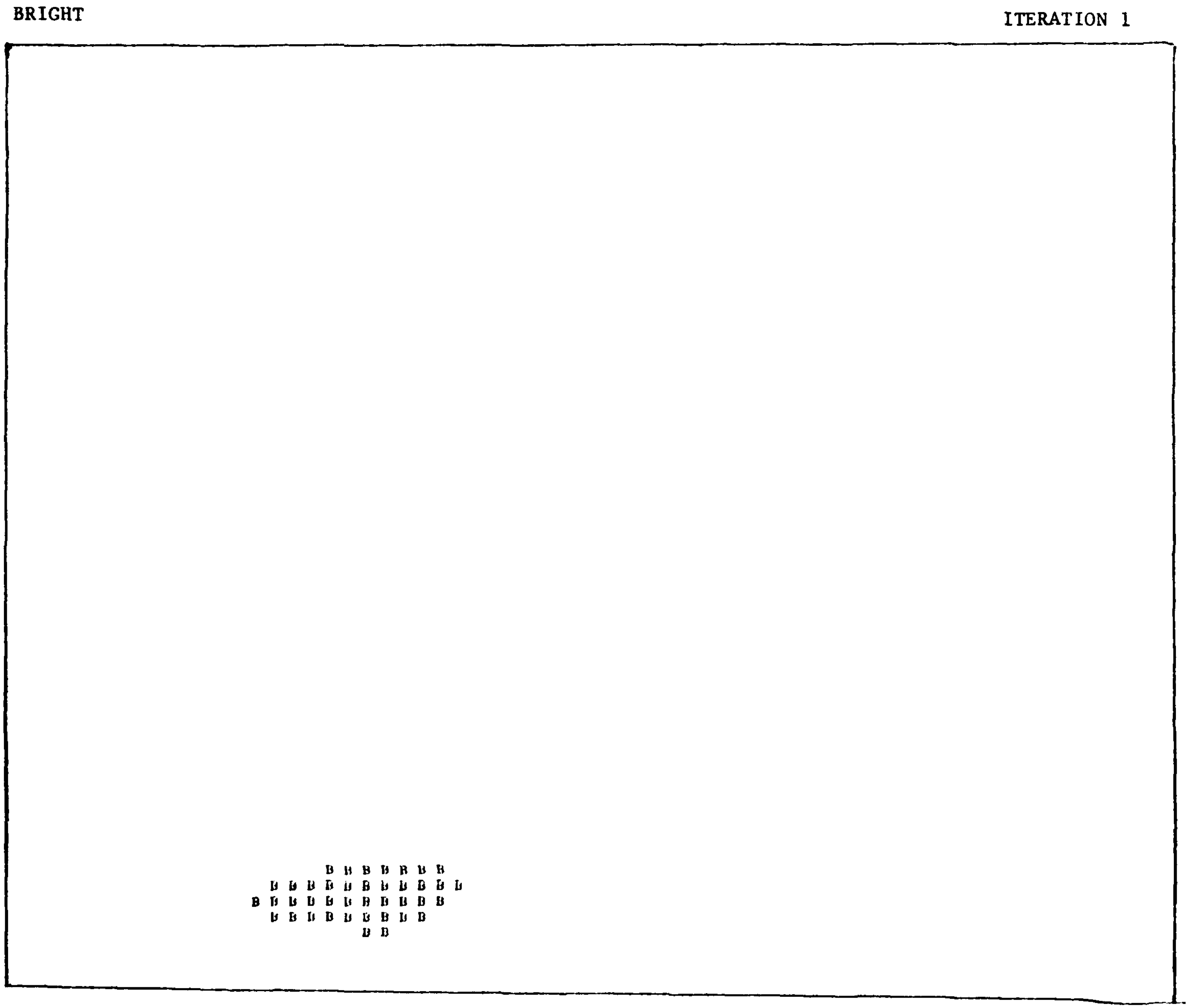


Figure 8b

BRIGHT

ITERATION 2

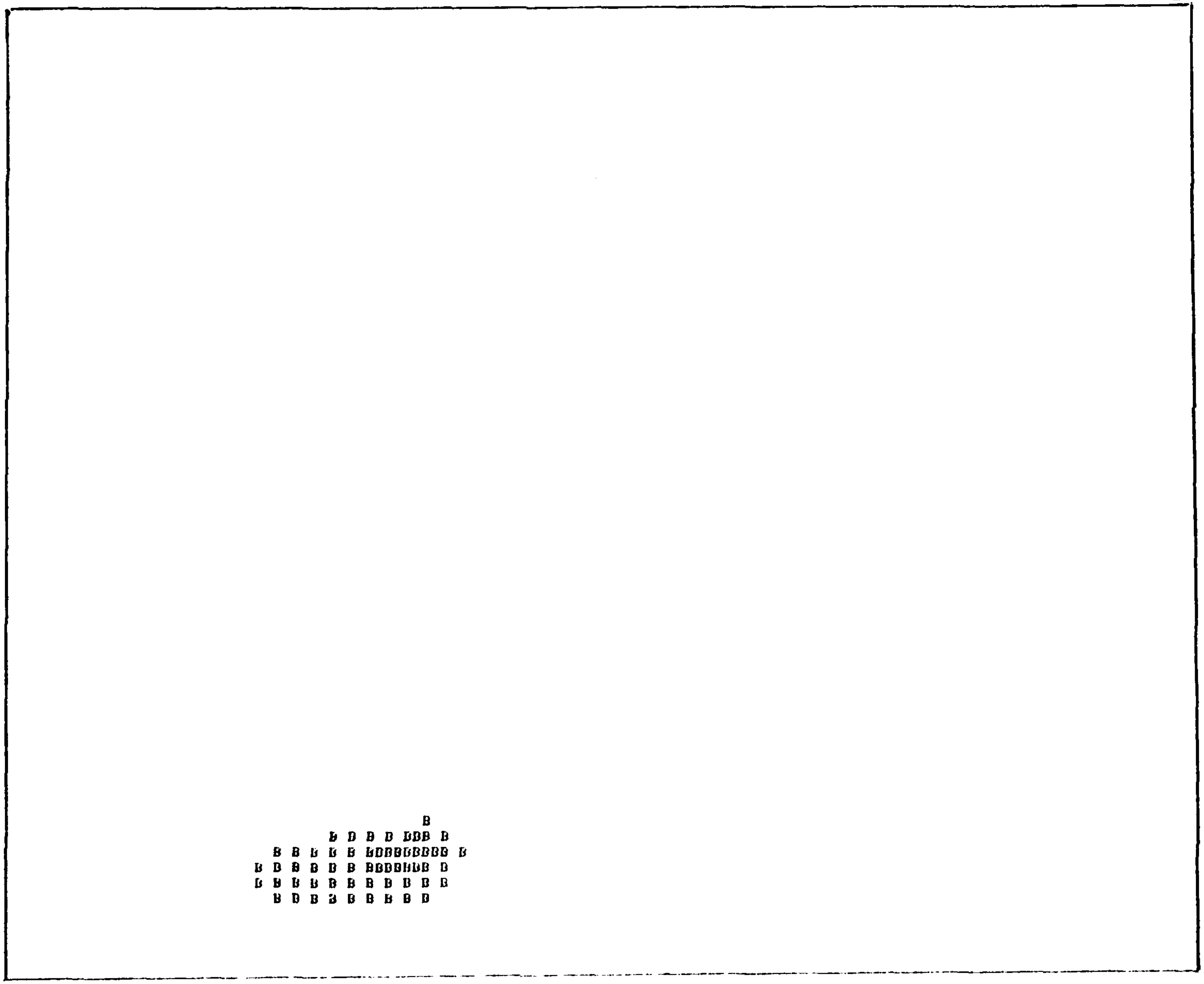


Figure 8c

BRIGHT

ITERATION 3

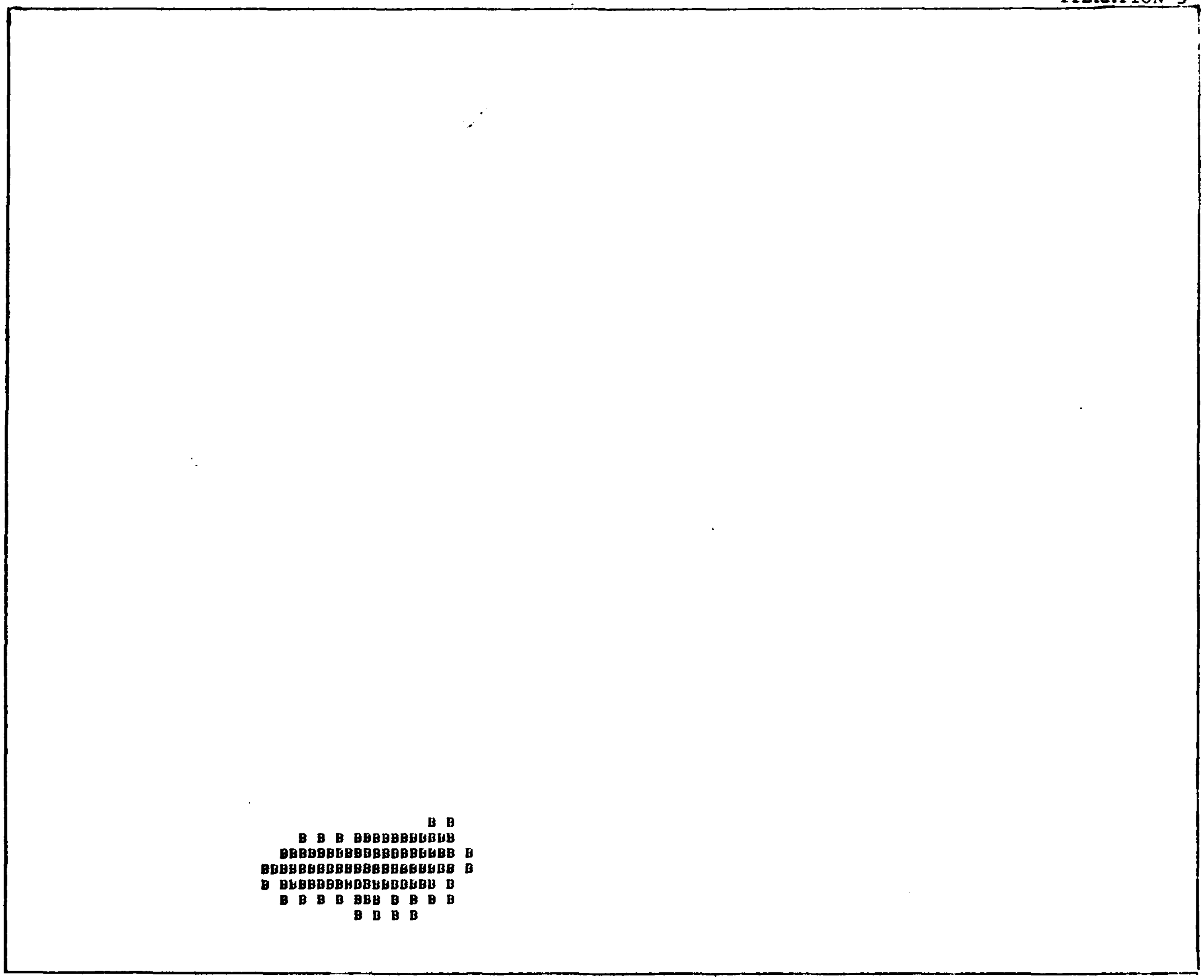


Figure 8d

BRIGHT

ITERATION 4

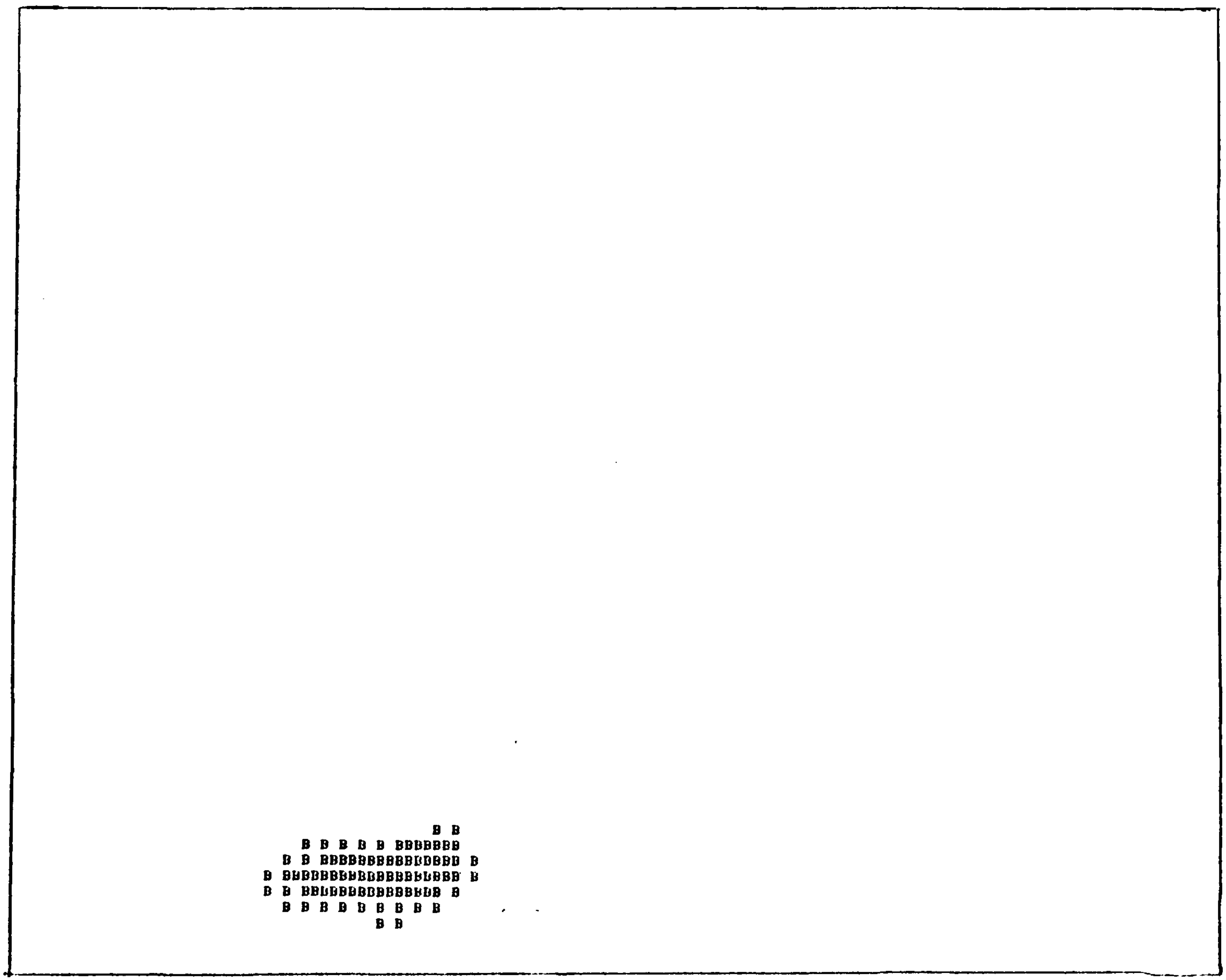


Figure 8e

BRIGHT

ITERATION 5

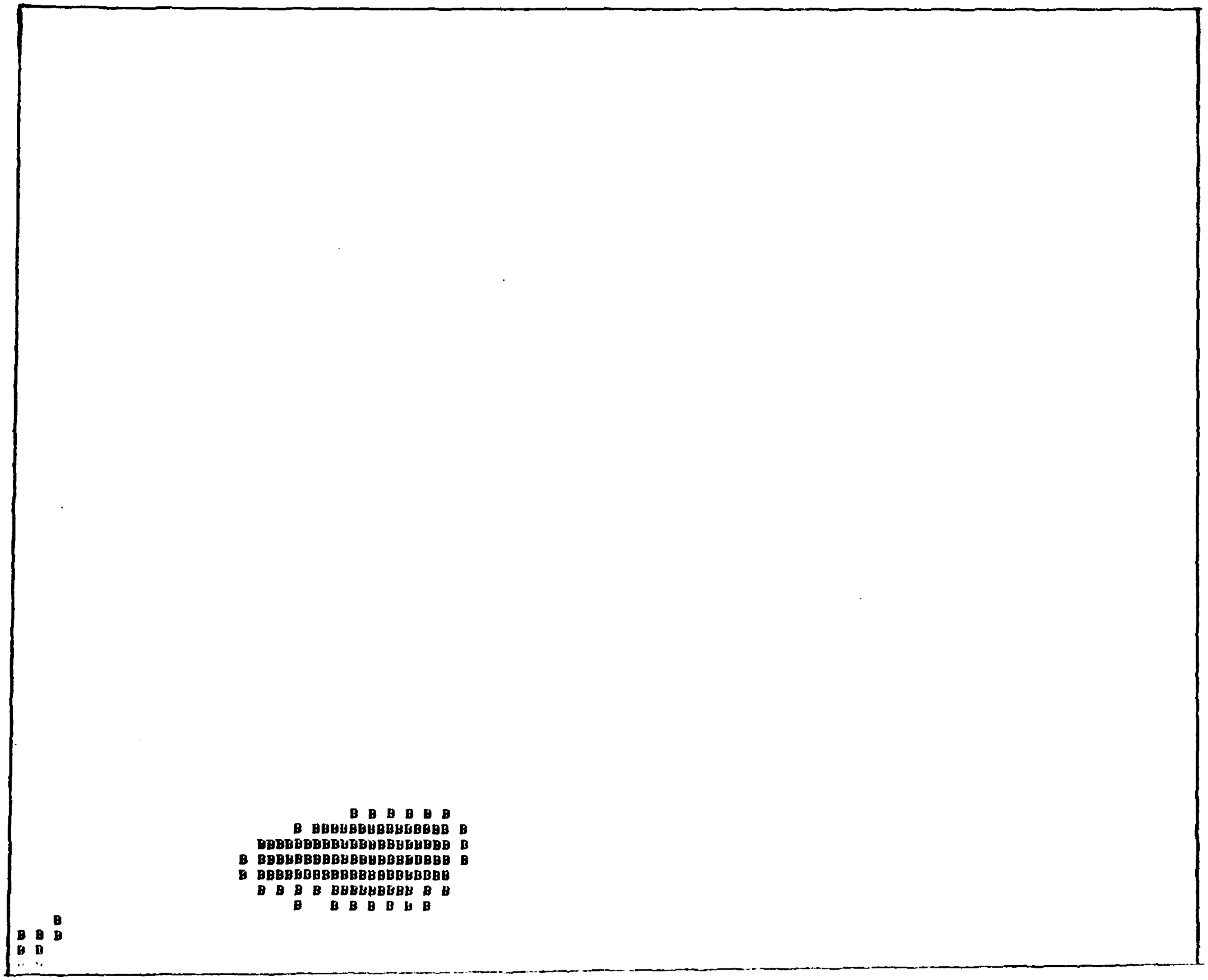


Figure 8f

BRIGHT

ITERATION 6

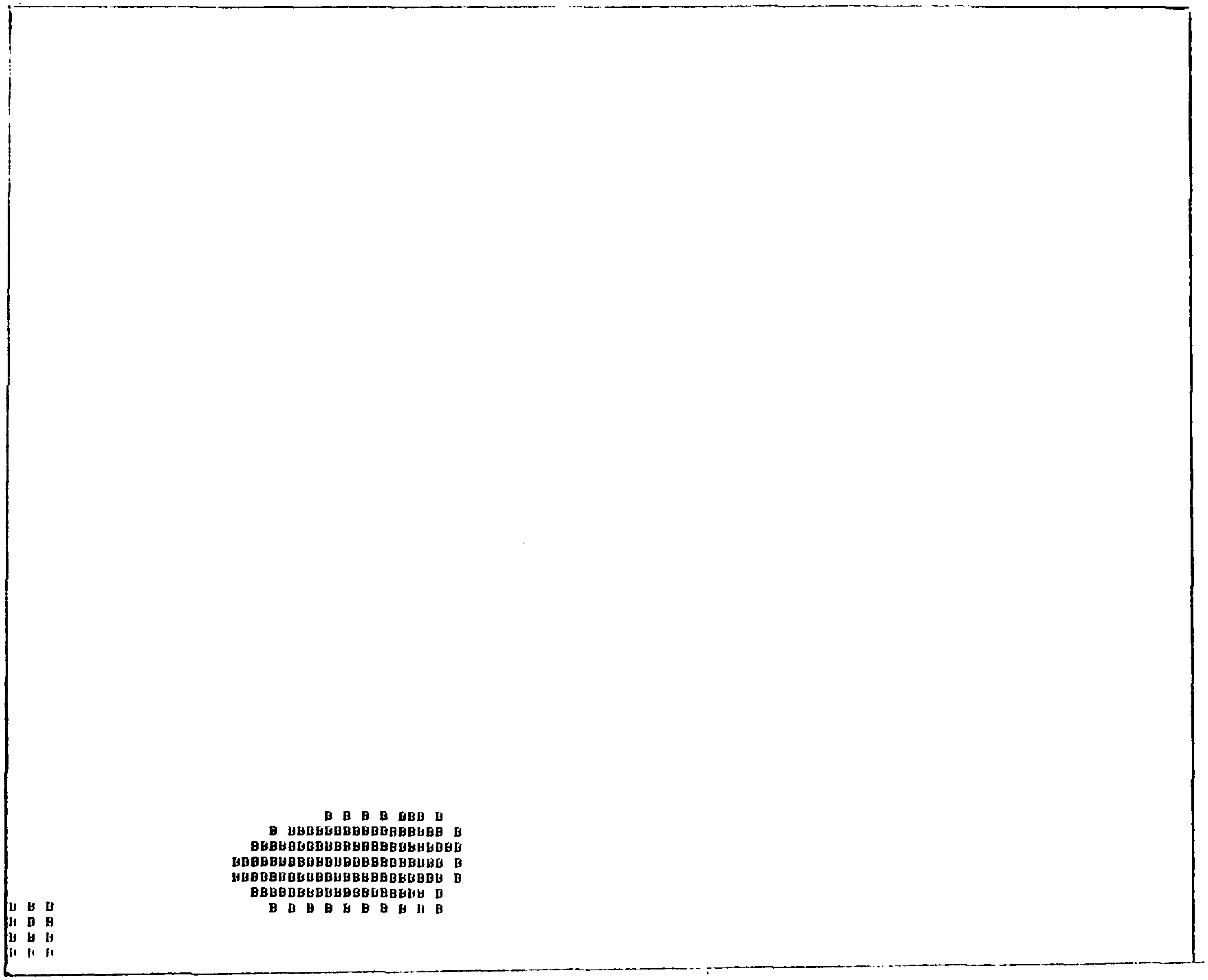


Figure 8g

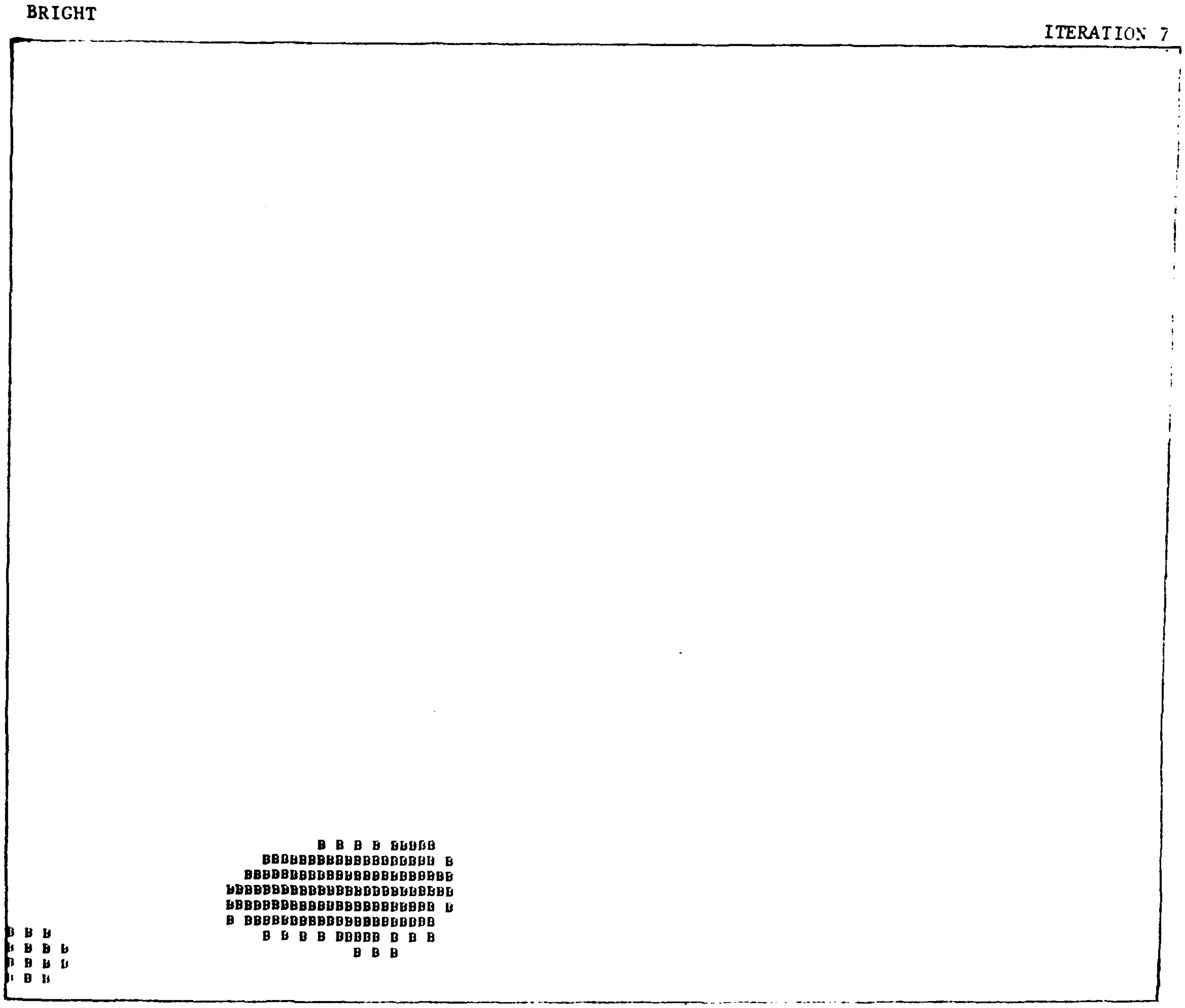


Figure 8h

BRIGHT

ITERATION 8

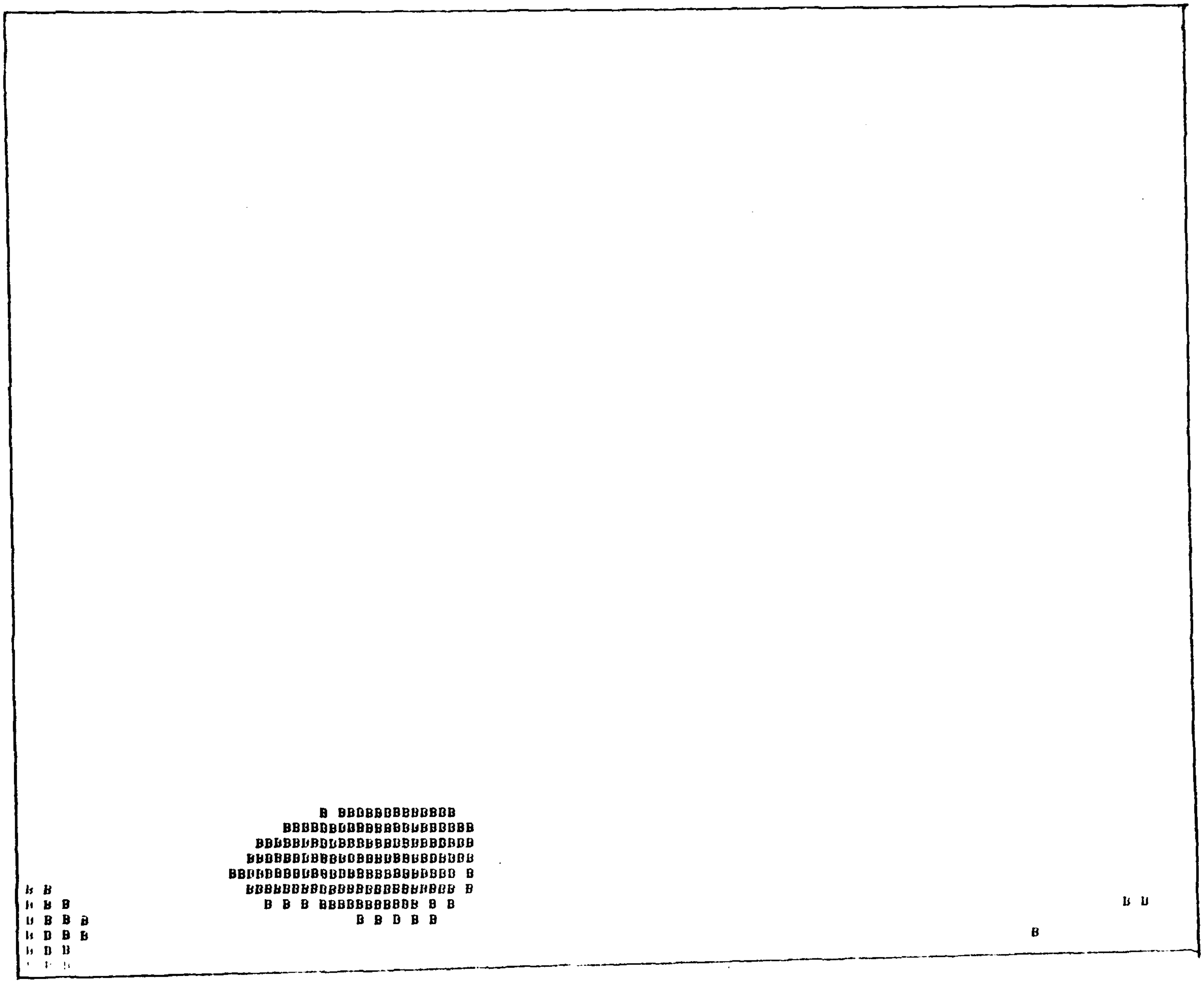


Figure 8i

BRIGHT

ITERATION 9

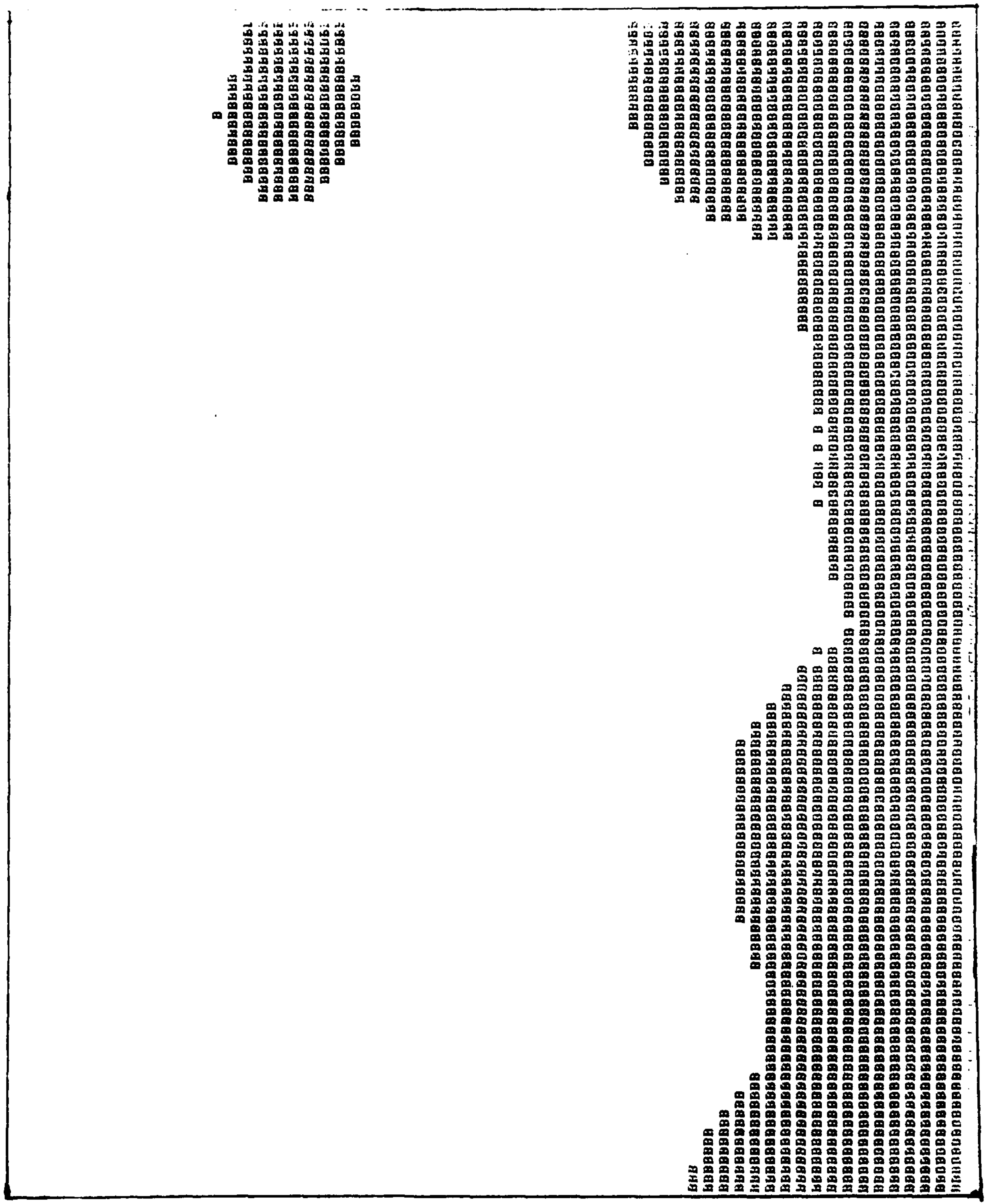


Figure 8j

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.07KBC= 0.78
KD= 0.00KCD= 0.00
KC= 0.20KBCD= 4.15
DELTA= 0.00
FB=G  .DIB=0,BIB=0.50
```

Figure 9

AD-A091 994 MARYLAND UNIV COLLEGE PARK COMPUTER SCIENCE CENTER F/G 9/2
ARCHITECTURE FOR HIGHER LEVEL DIGITAL IMAGE PROCESSING.(U)
FEB 80 A HELLAND, J HUNG DAAG53-76-C-0138
UNCLASSIFIED NL
2 of 2

END
DATE
FILMED
81
DTIC

BRIGHT

ITERATION 0

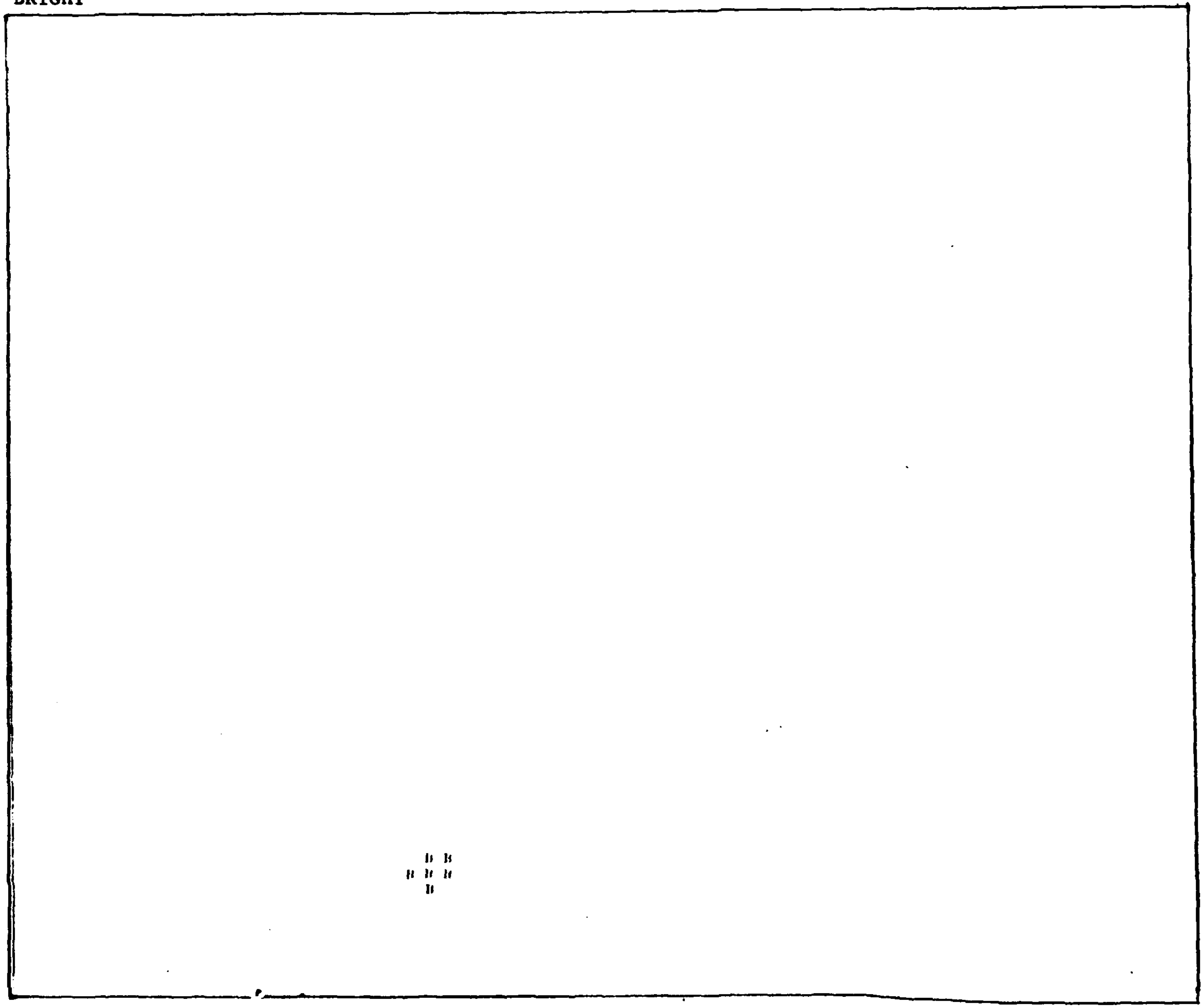


Figure 9a

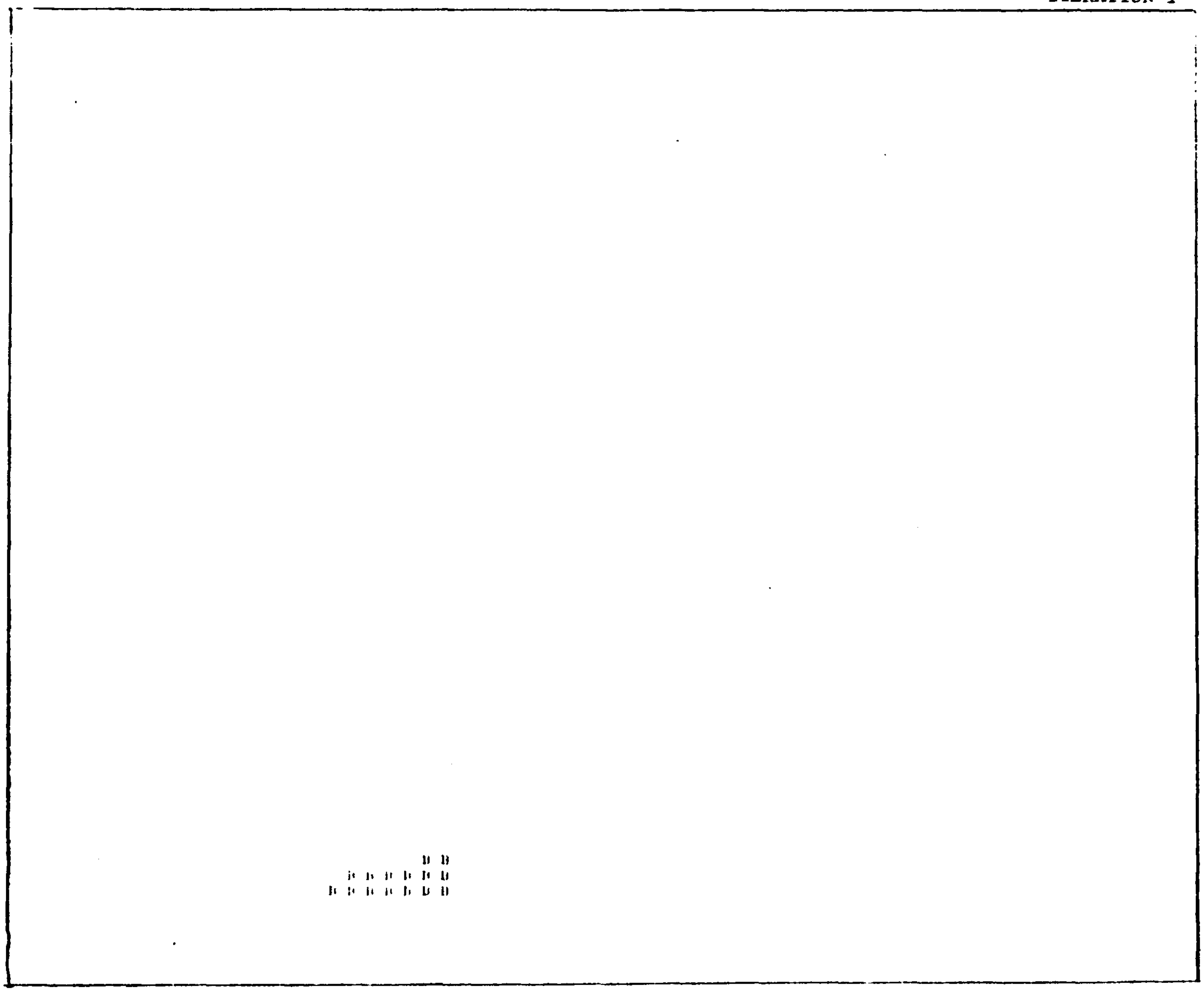


Figure 9b

BRIGHT ITERATION 2

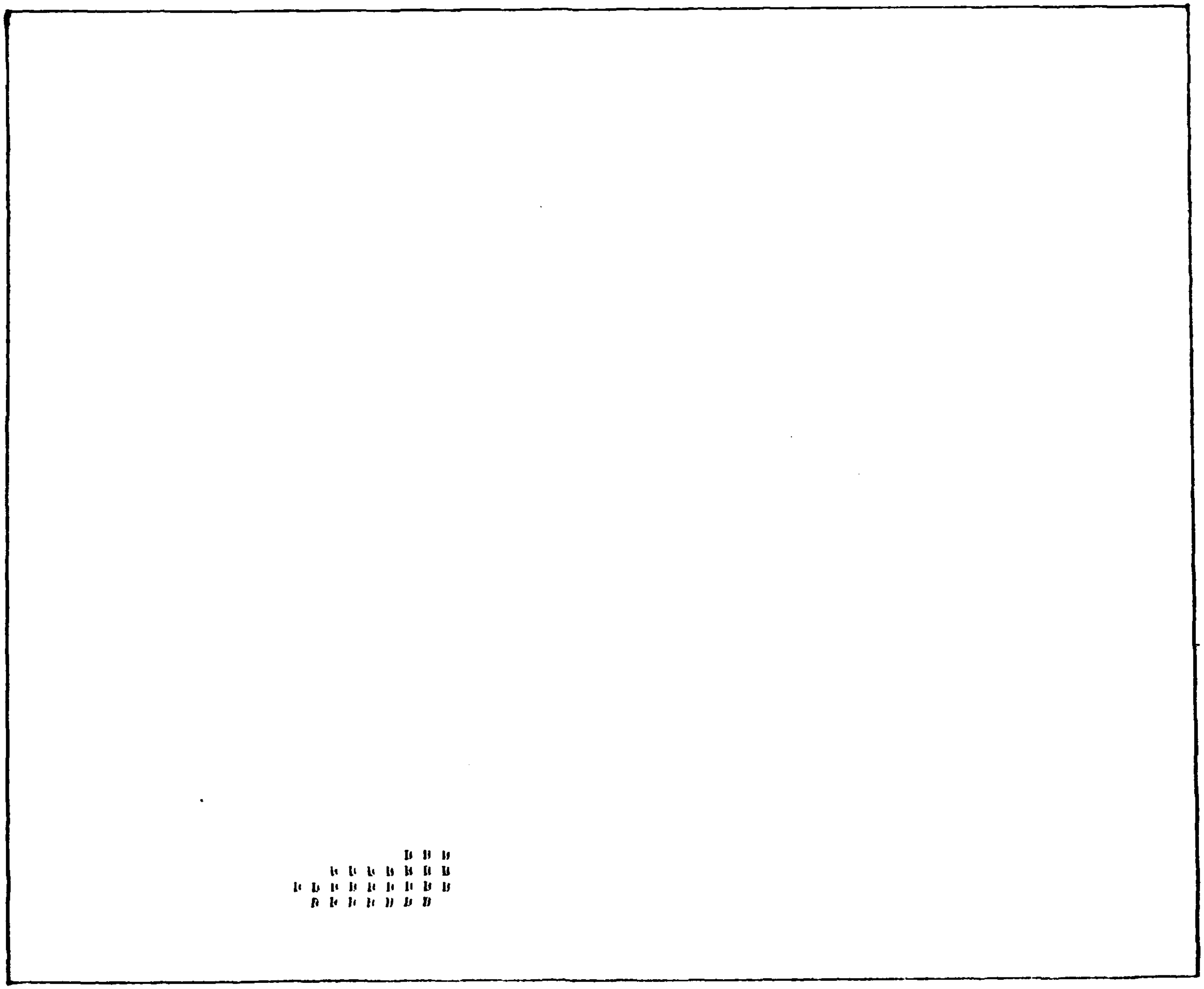


Figure 9c

BRIGHT ITERATION 3

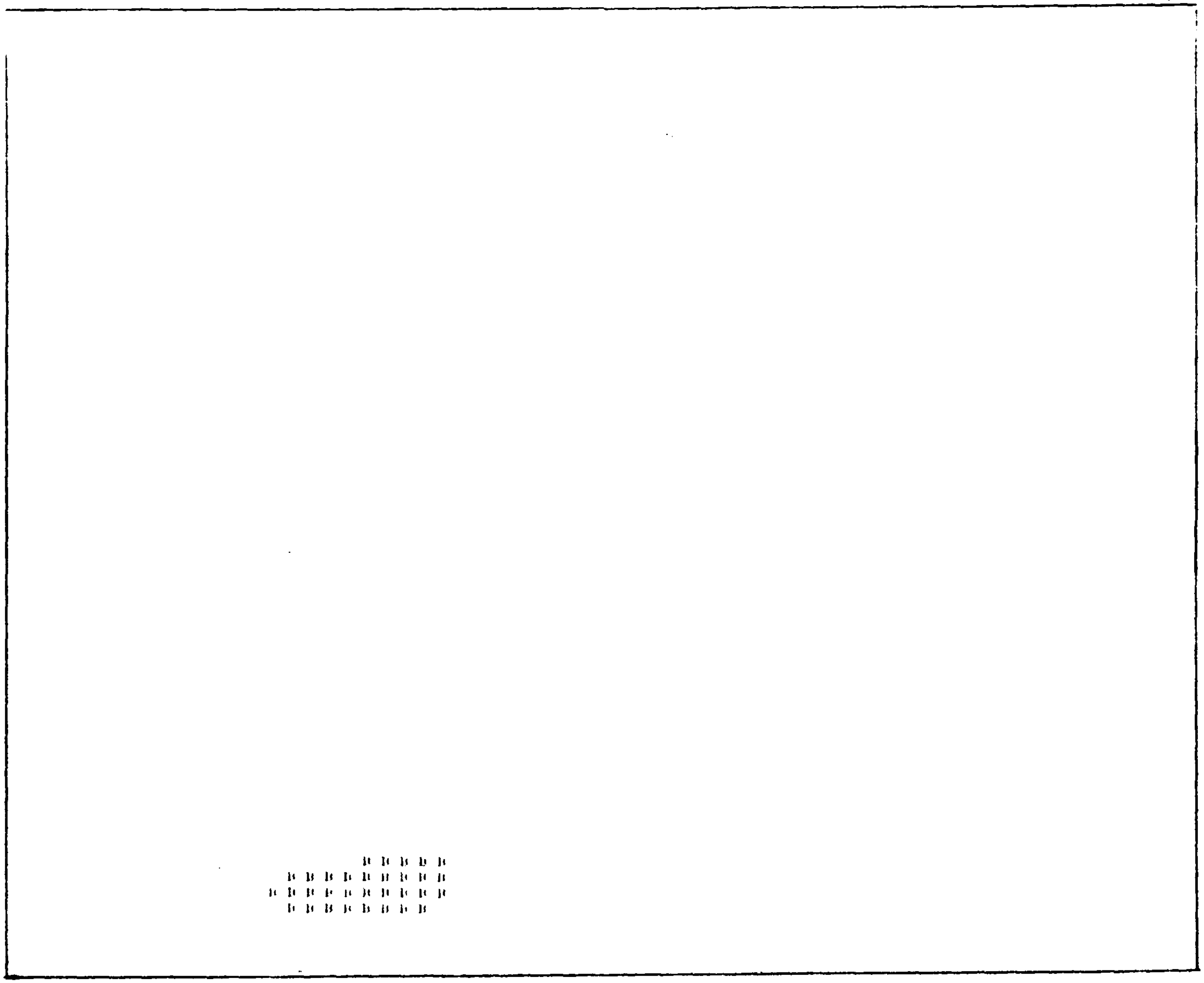


Figure 9d

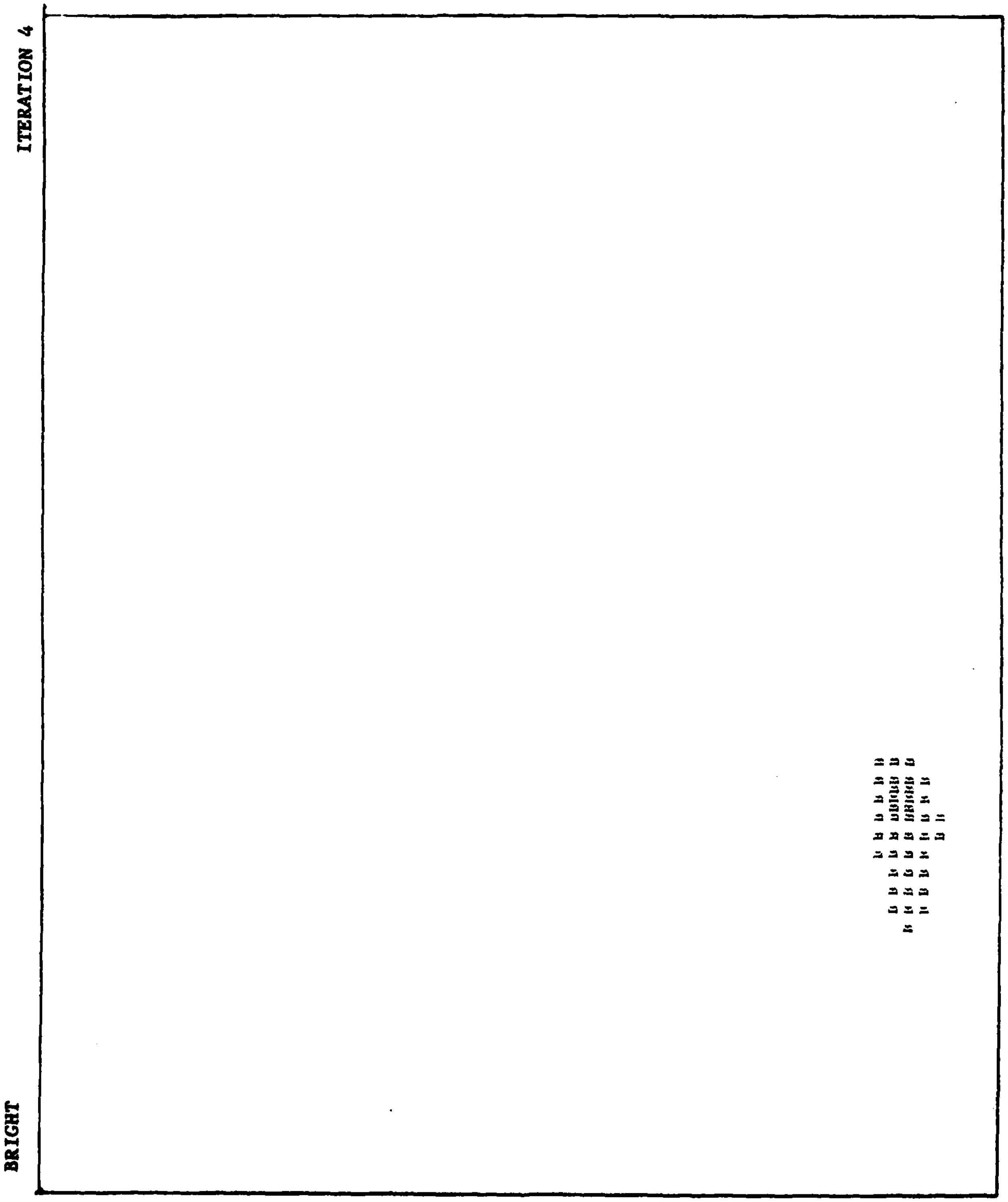


Figure 9e

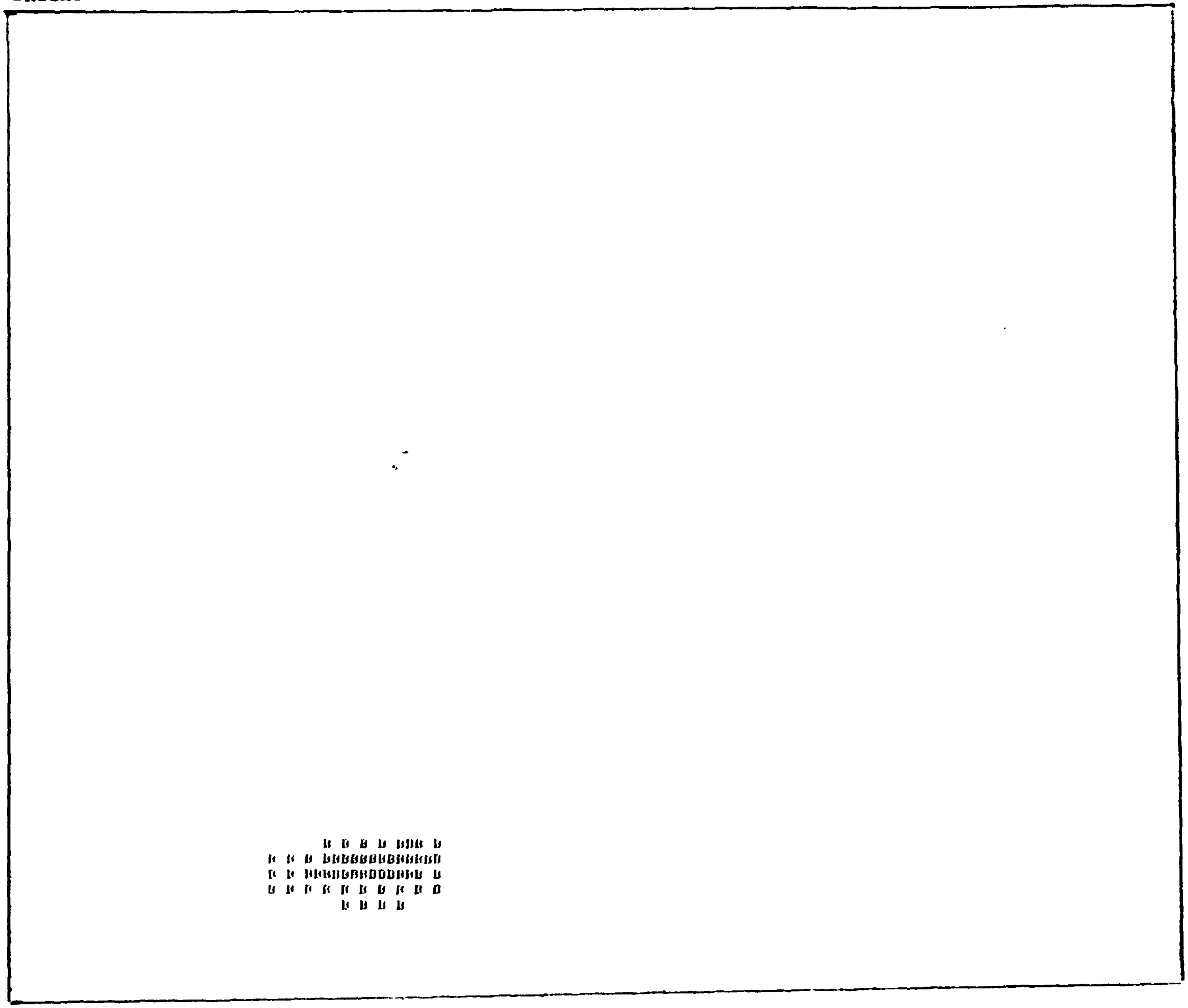


Figure 9f

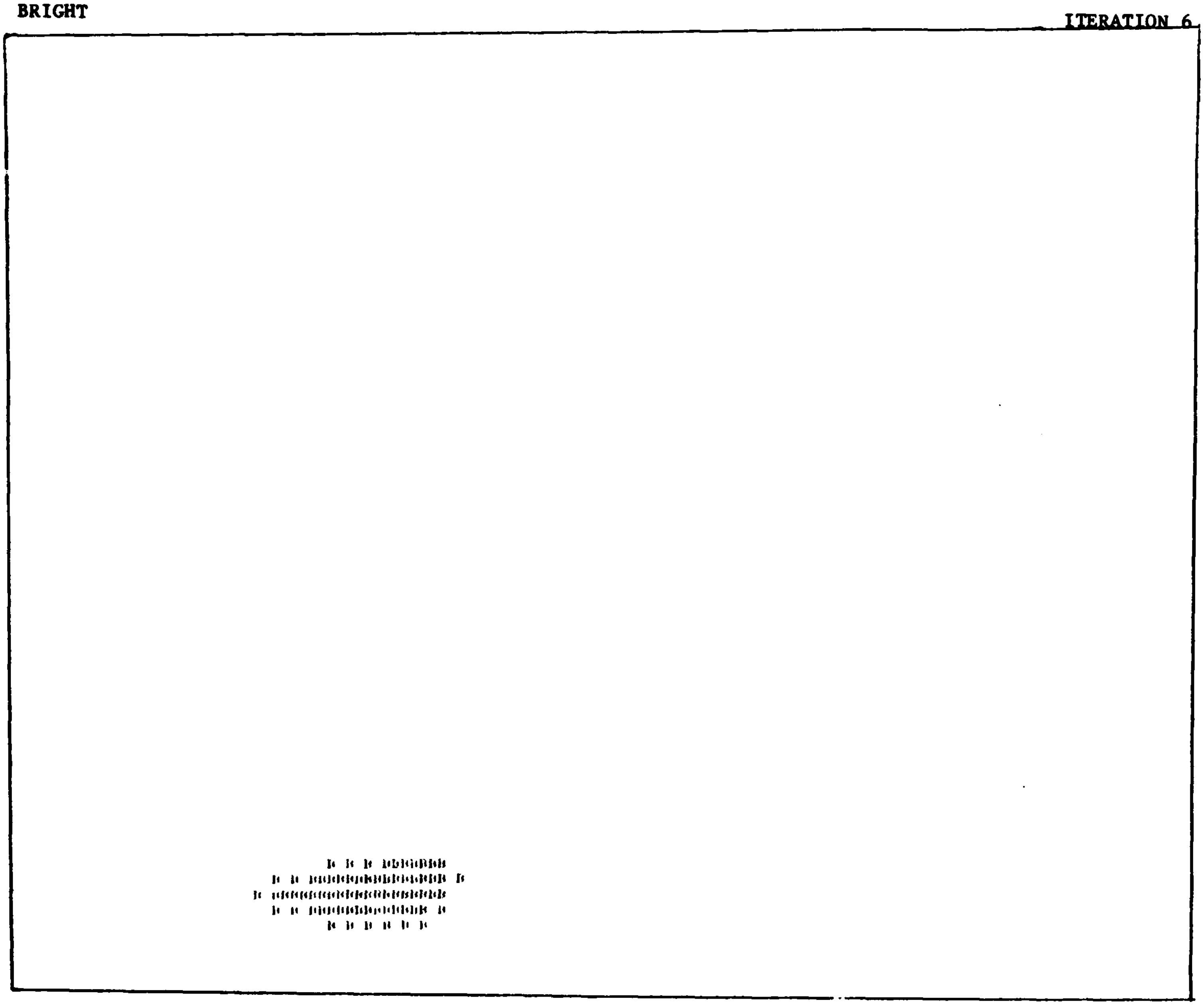


Figure 9g

BRIGHT

ITERATION 7

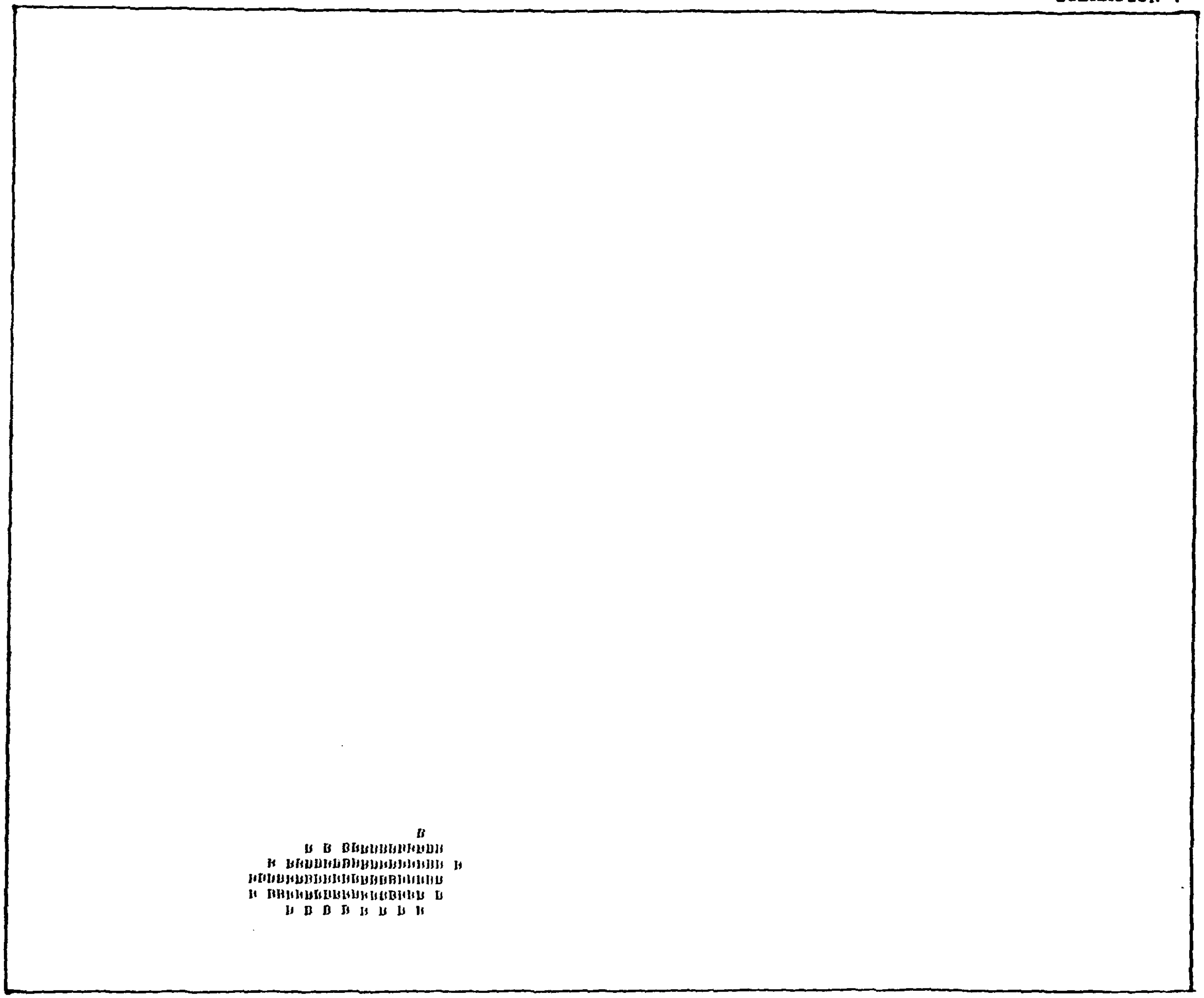


Figure 9h

BRIGHT

ITERATION 8

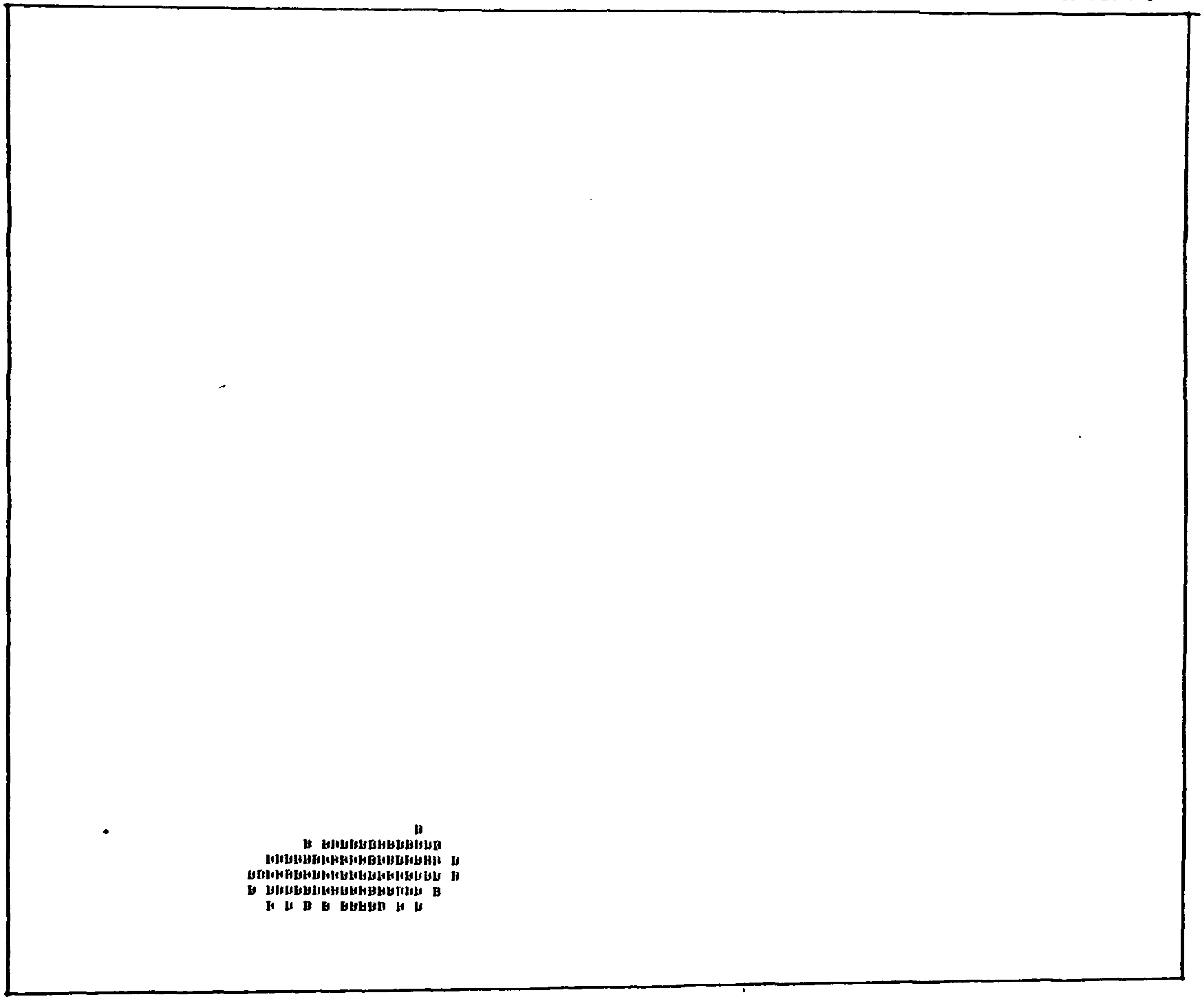
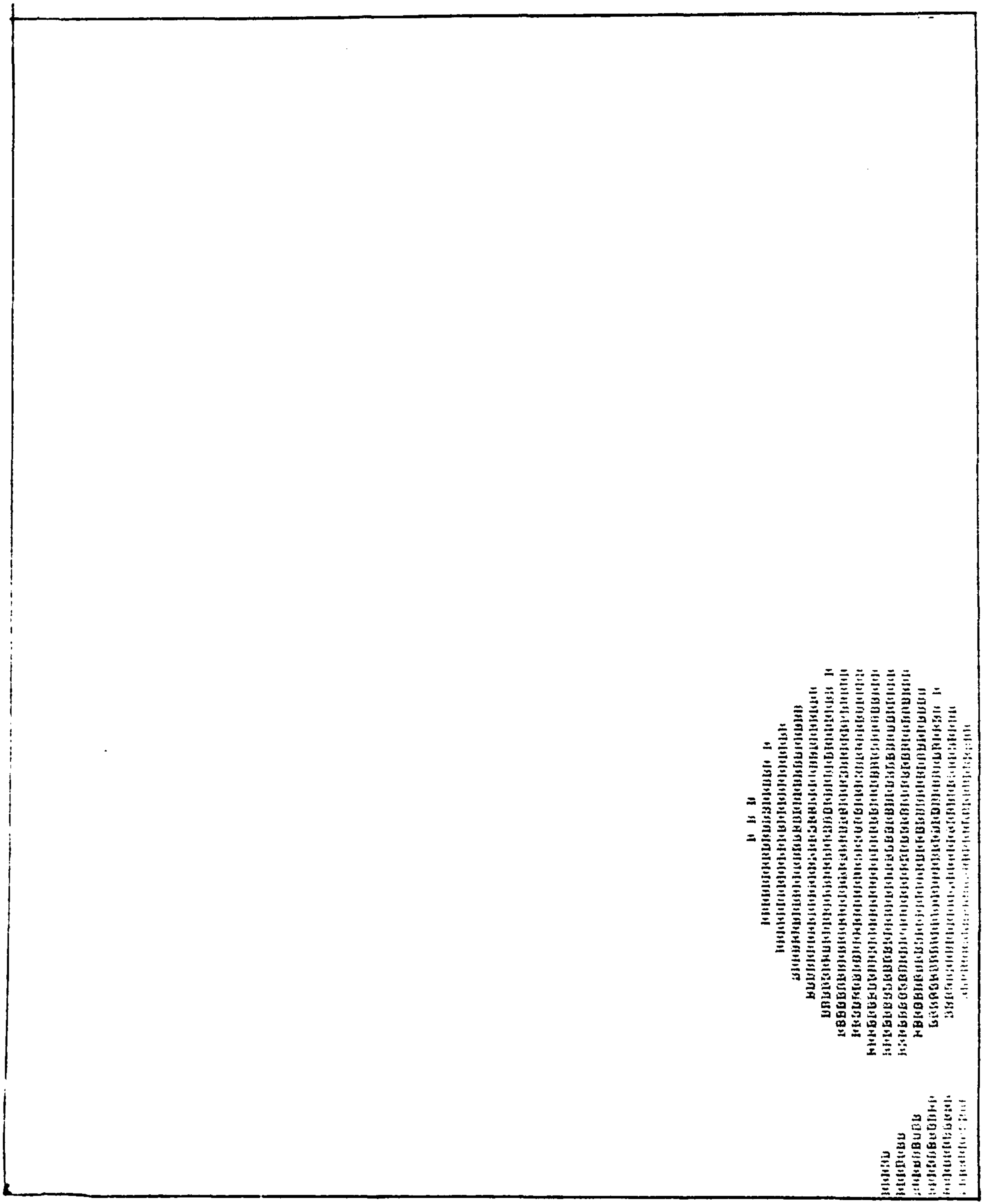


Figure 9i

BRIGHT ITERATION 100



```
FILE NAME= TESTP2. DAT
BRIGHT/DARK/CLUTTER RELAXATION 32 BY 32
KB= 1.00KBC= 0.00
KD= 1.00KCD= 0.00
KC= 1.00KBCD= 0.00
DELTA= 0.50
NUM= 1
BRIGHT
```

Figure 10

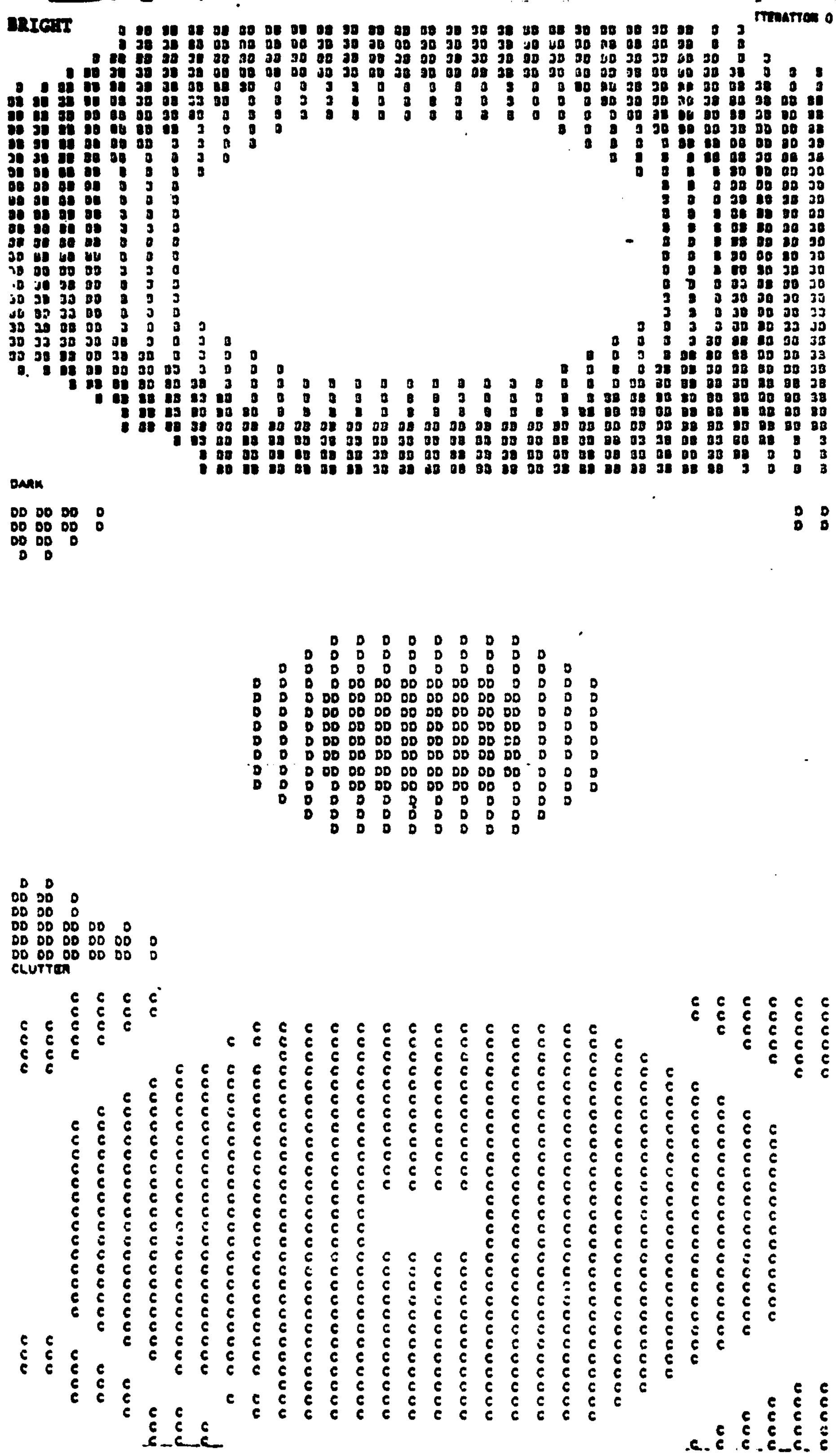


Figure 10a

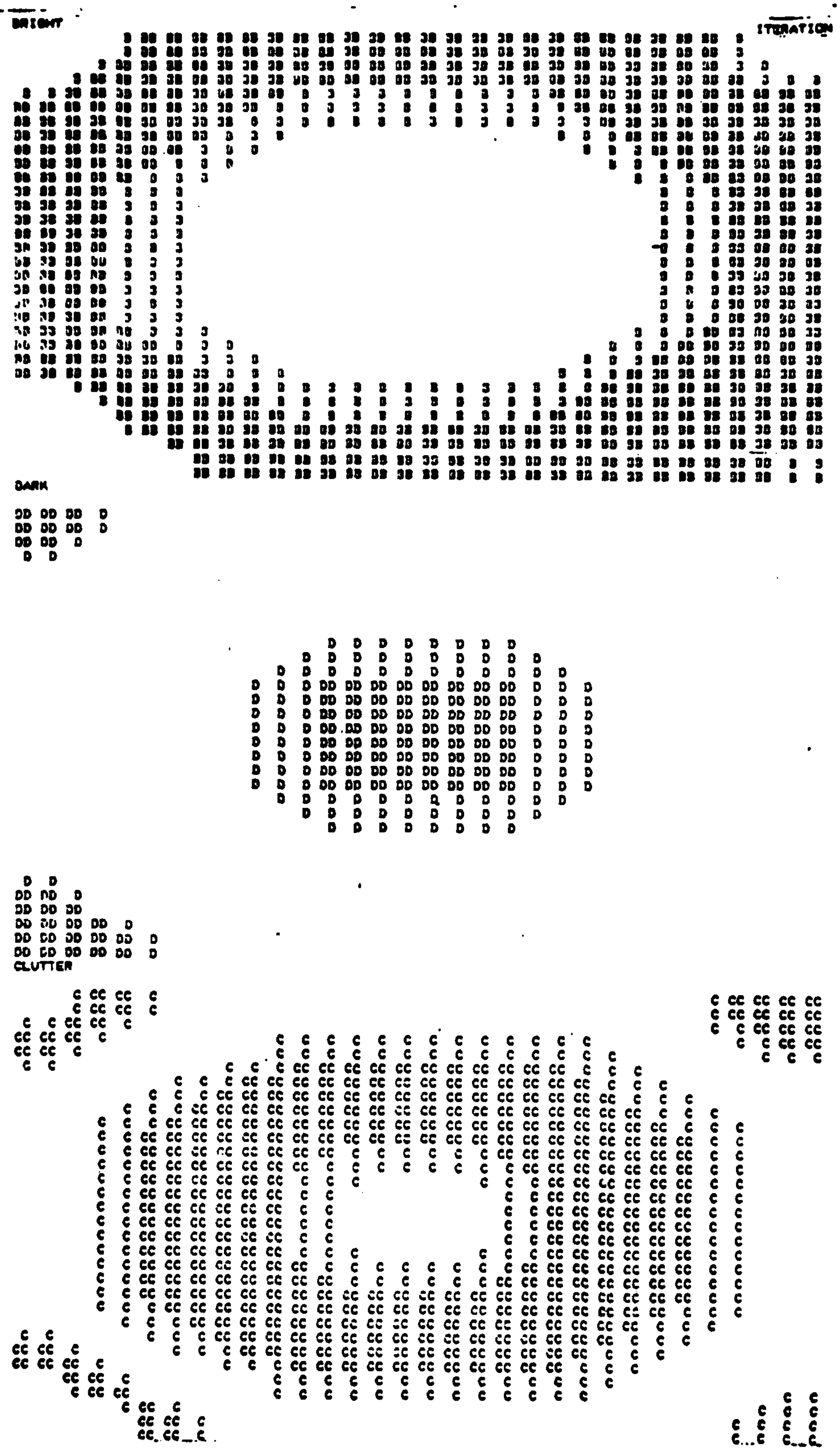


Figure 10b

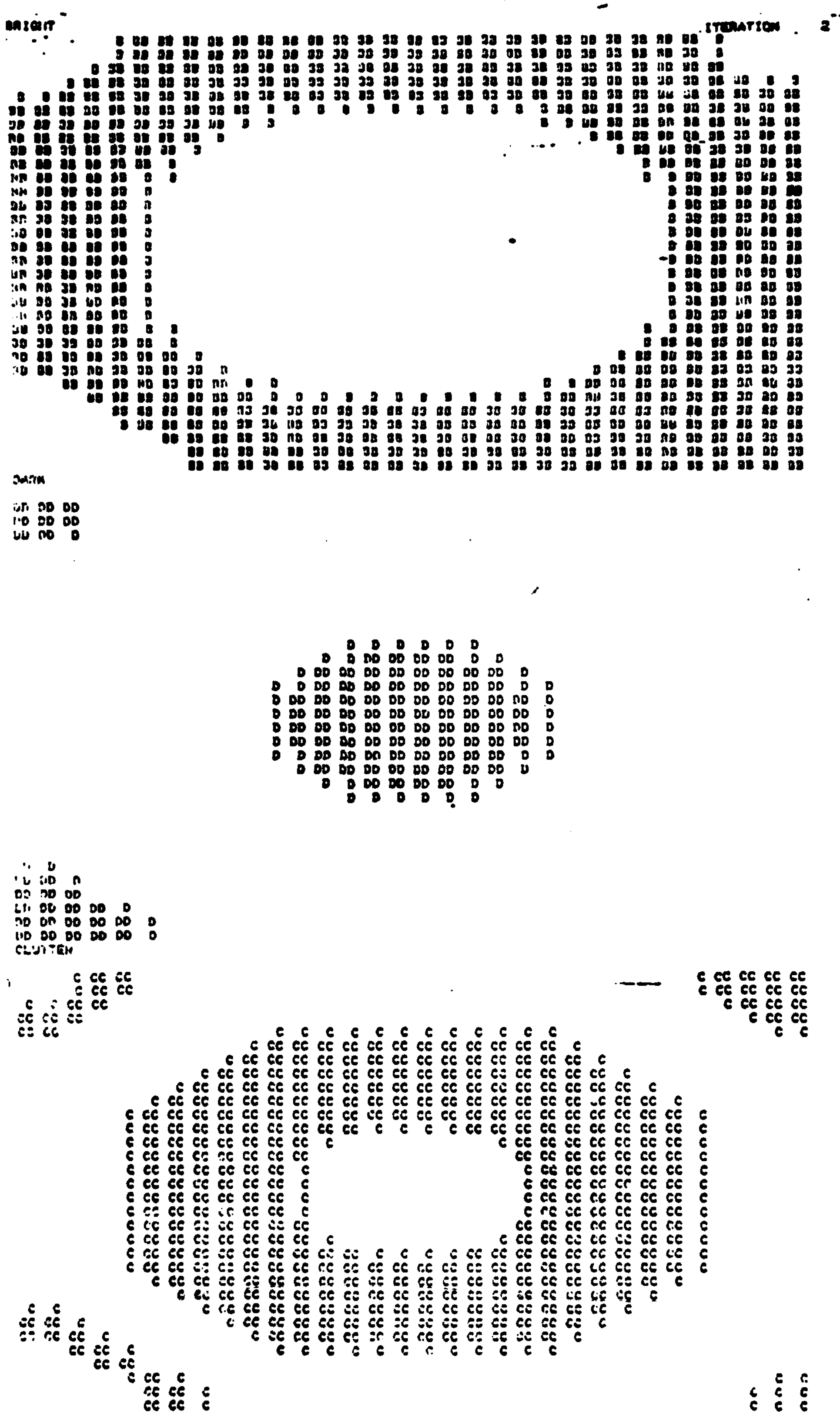


Figure 10c

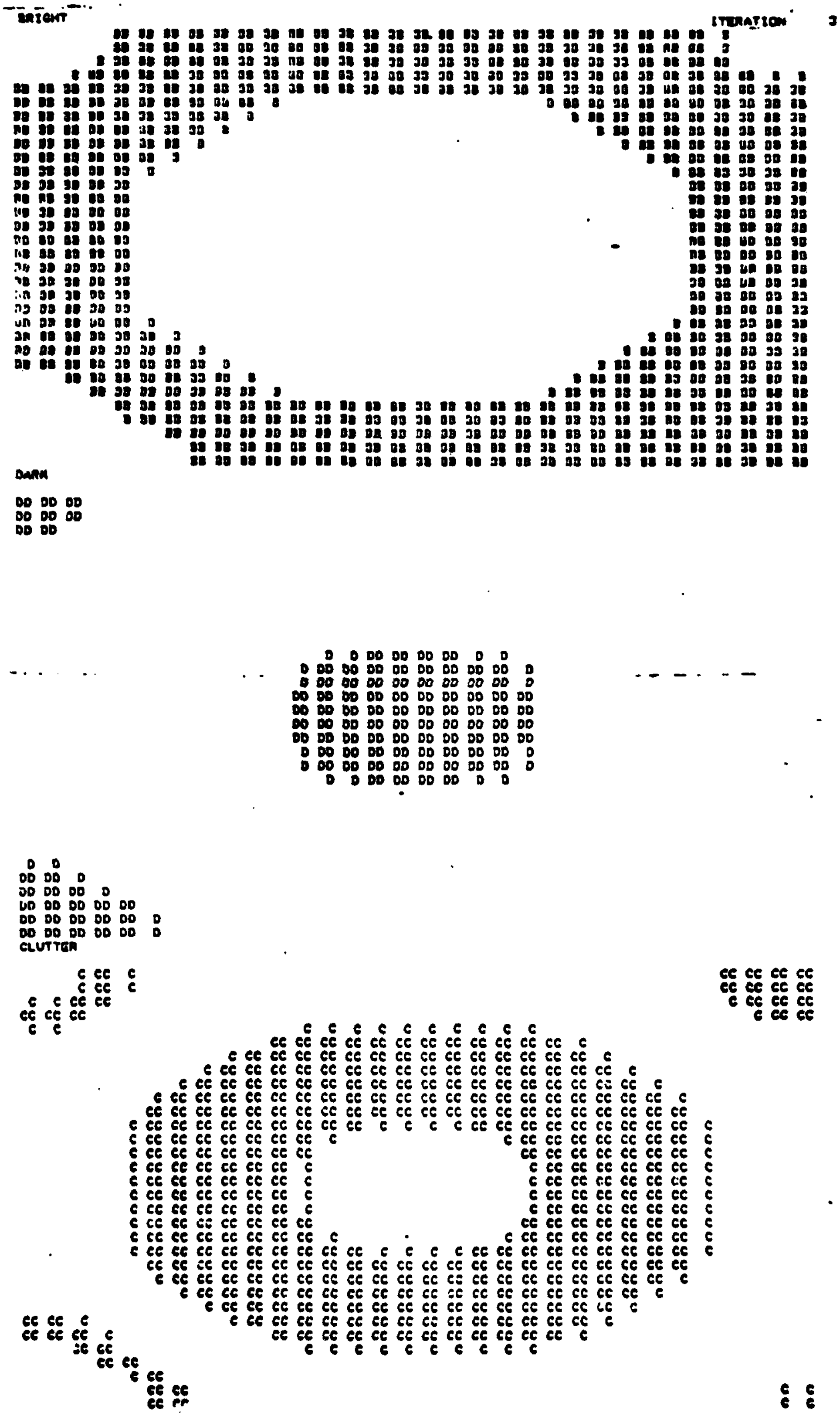


Figure 10d

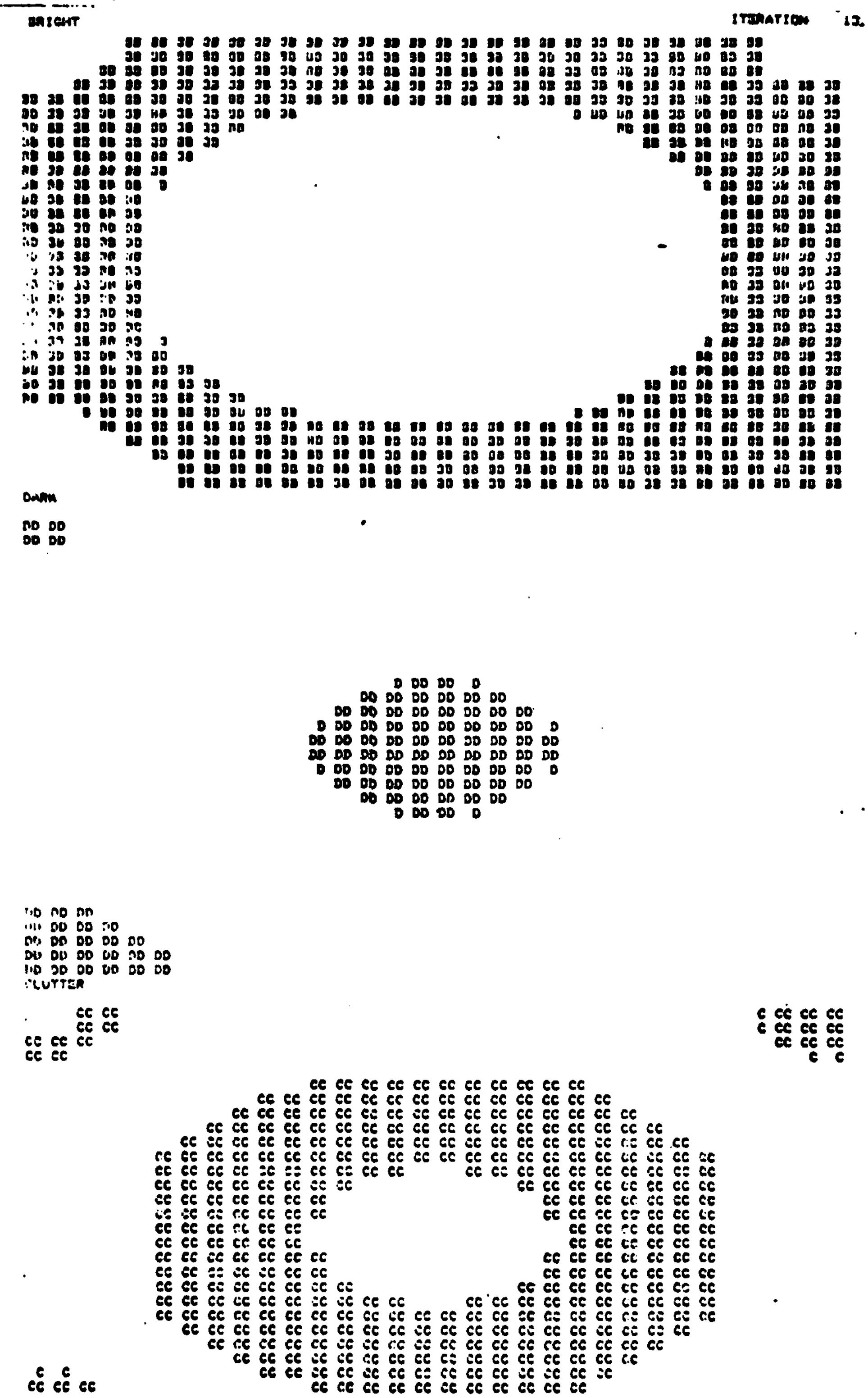


Figure 10e

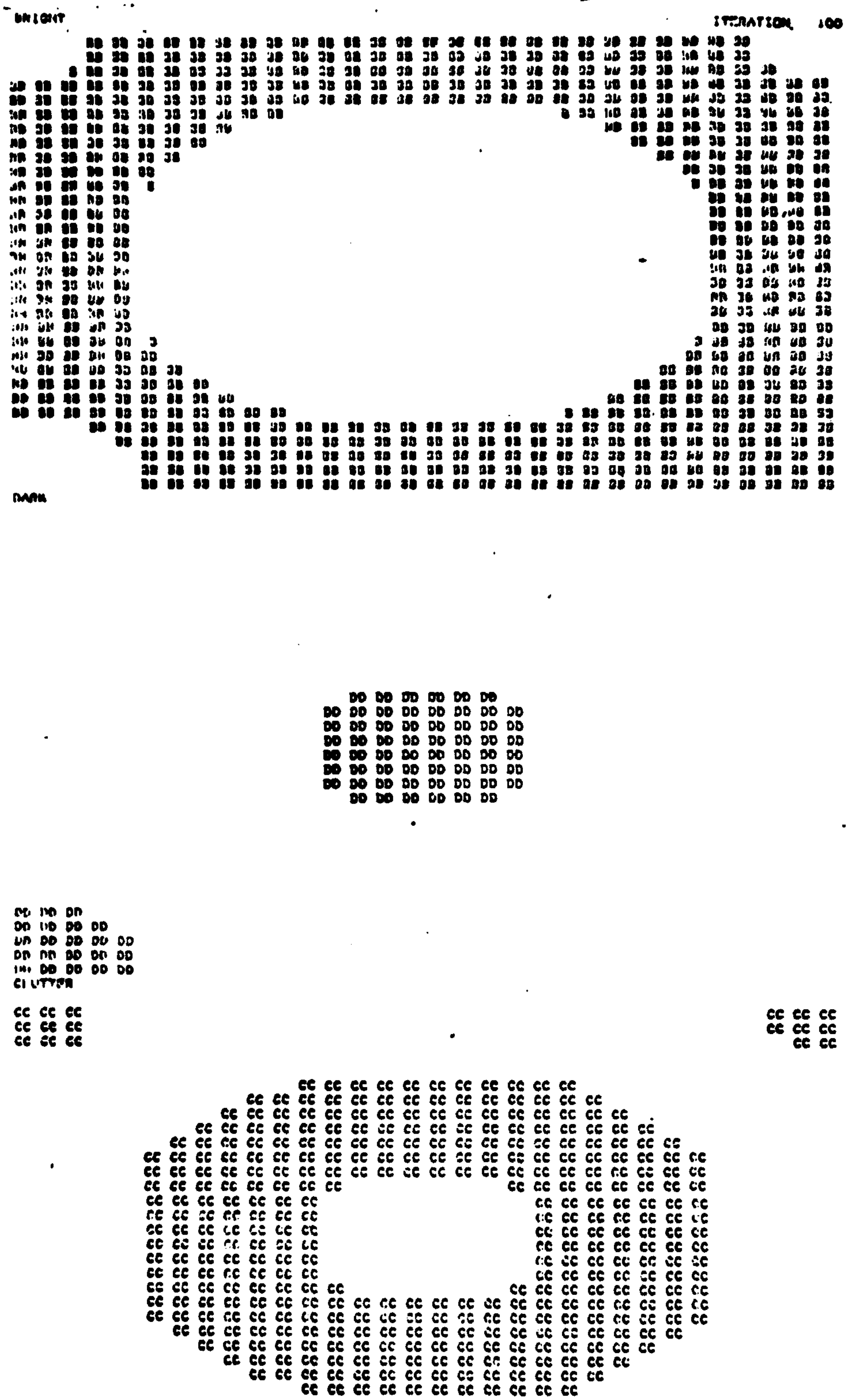


Figure 10f

```
FILE NAME= UMPAT1.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64BY 64)
KB= 1.00KBC= 0.00
KD= 1.00KCD= 0.00
KC= 1.00KBCD= 0.00
DELTA= 0.50
```

Figure 11

BRIGHT

ITERATION 0

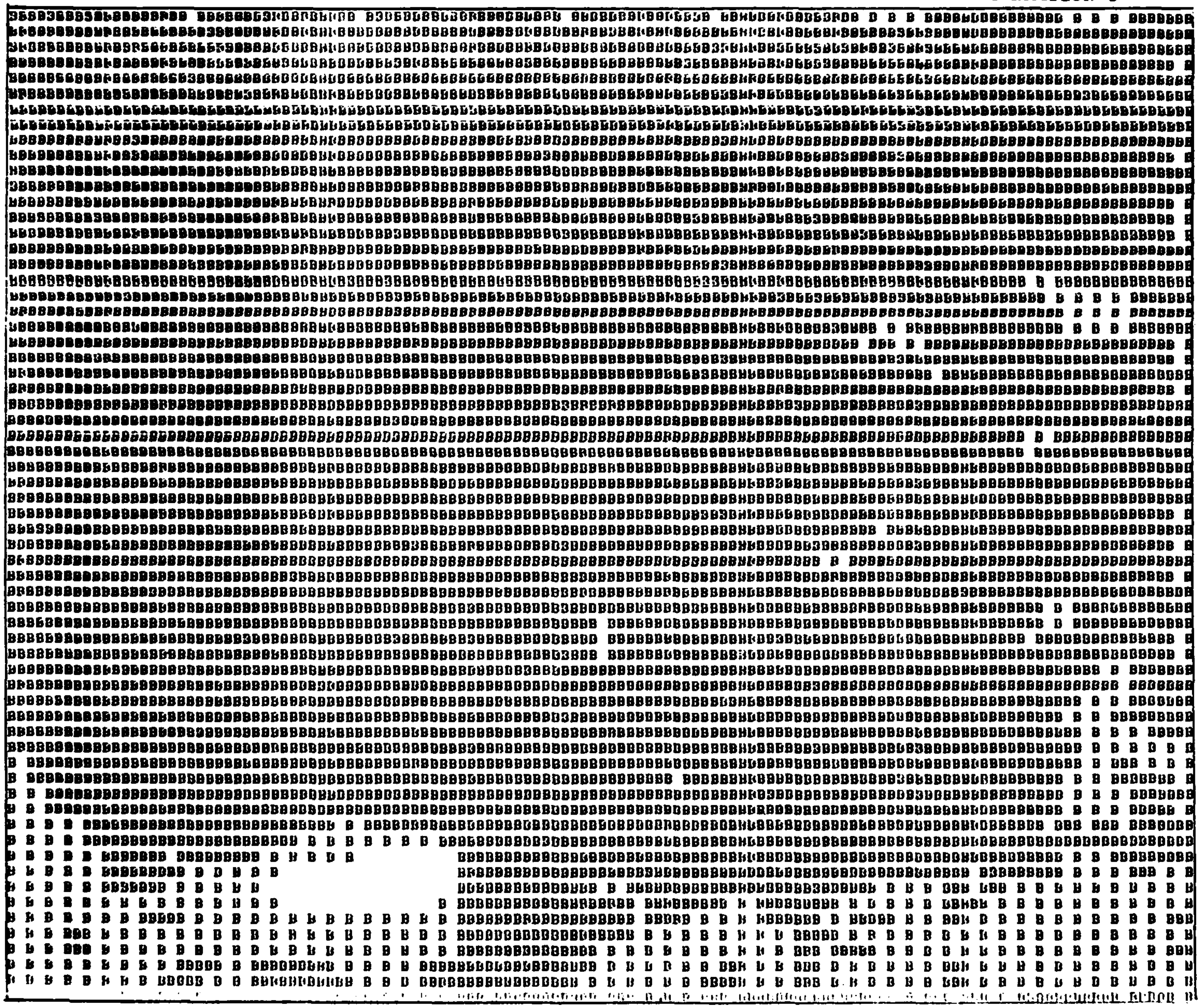


Figure 11a

DARK

ITERATION 0

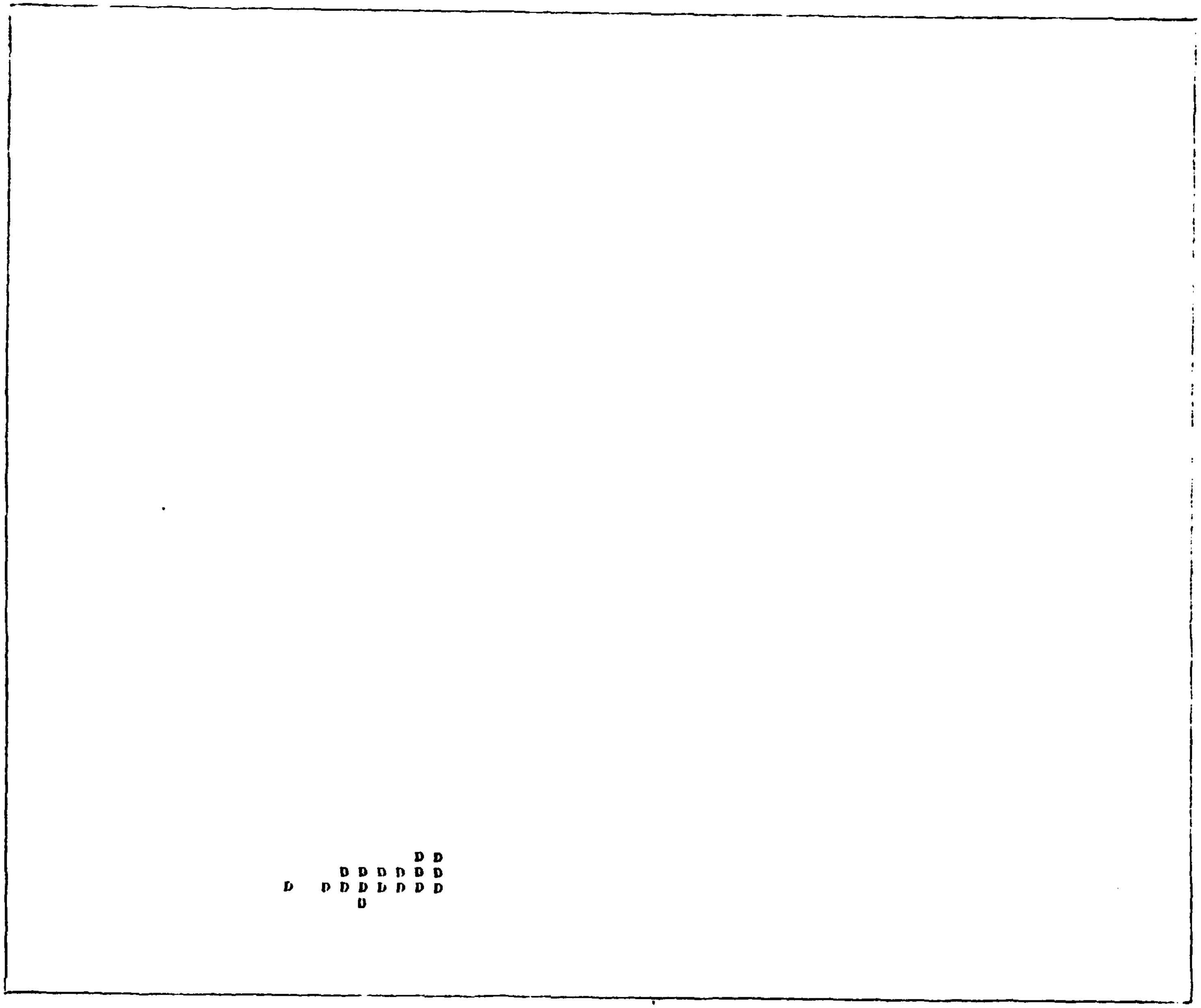


Figure 11b

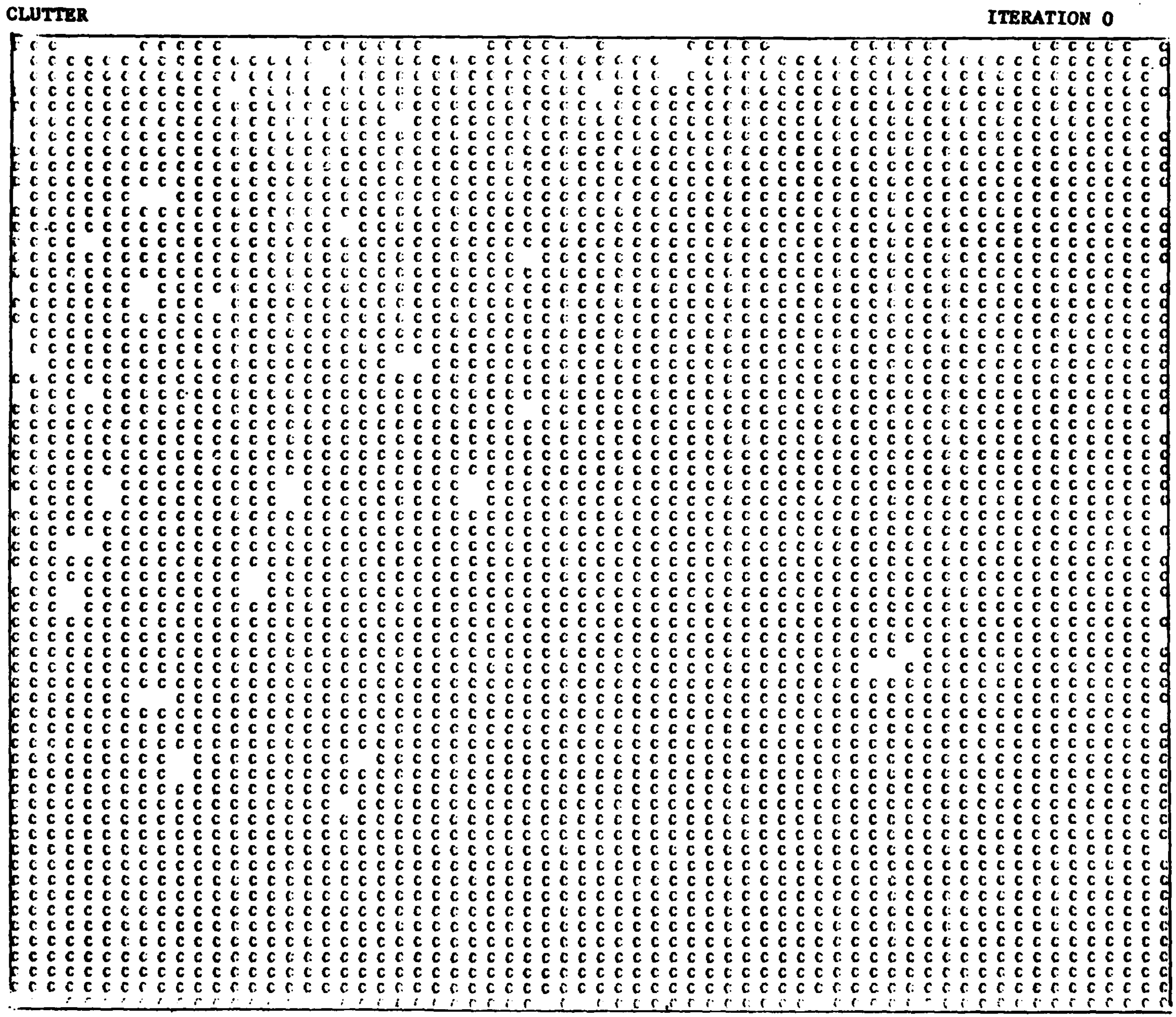


Figure 11c

BRIGHT ITERATION 1

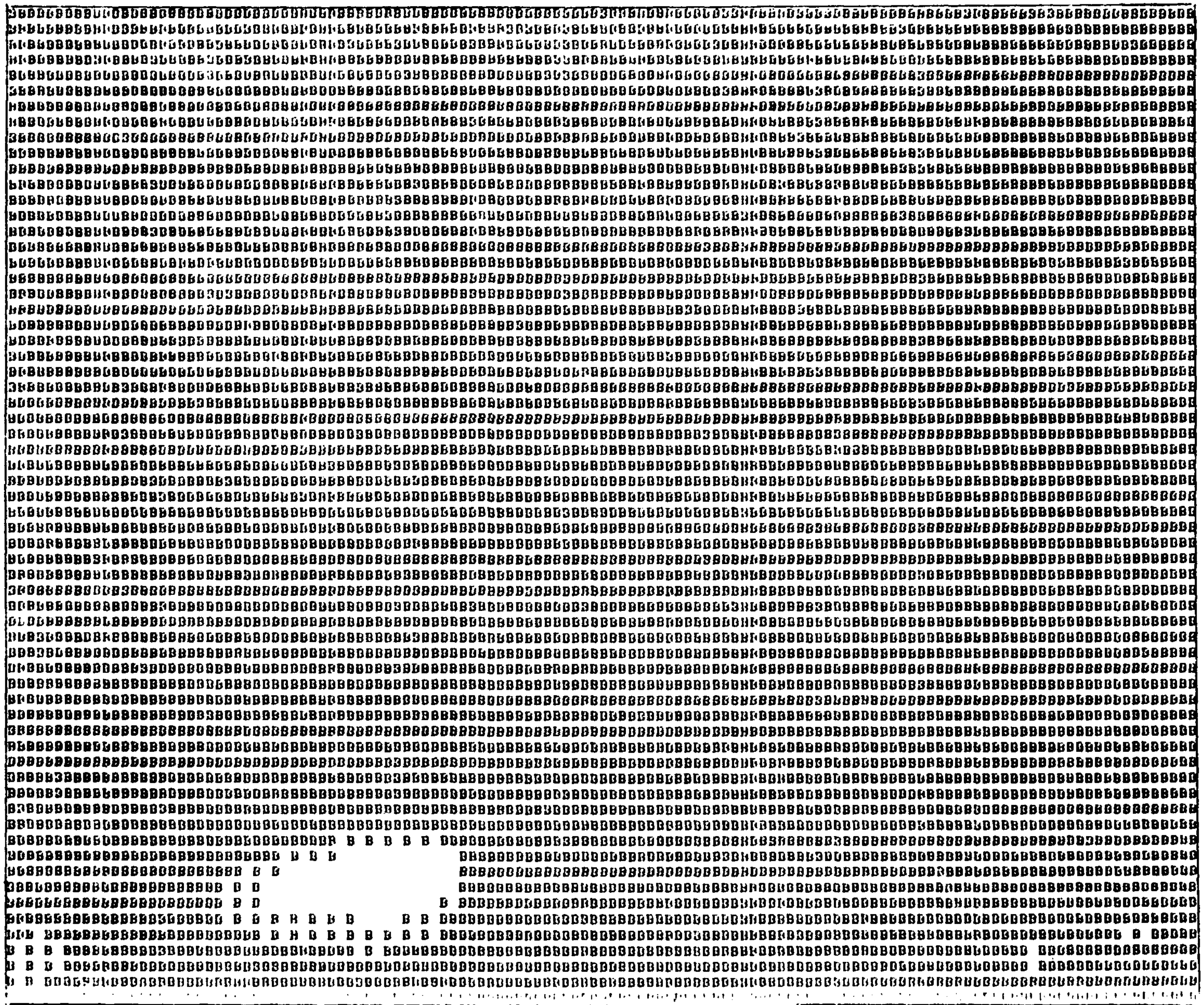


Figure 11d

DARK ITERATION 1

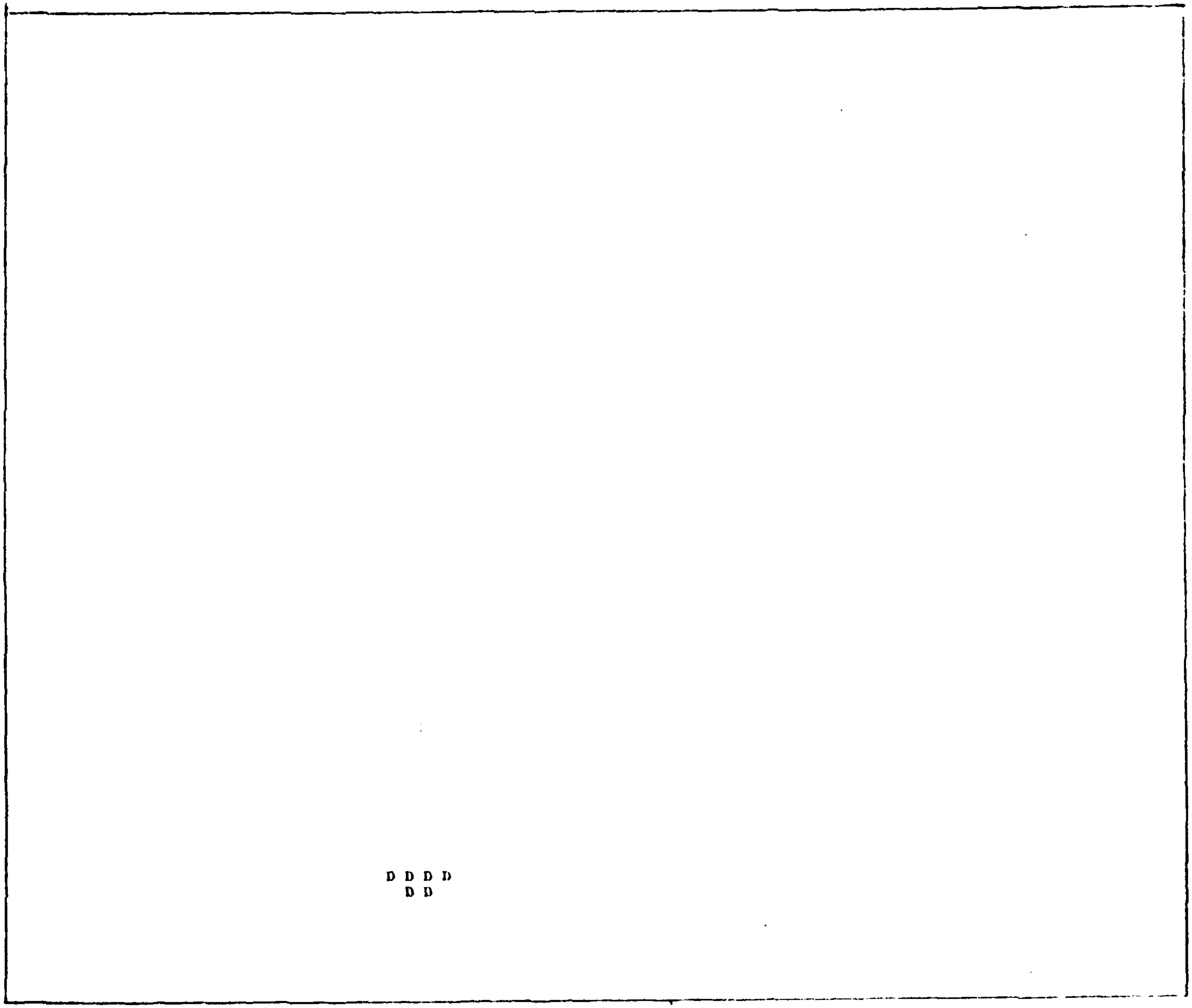


Figure 11e

Figure 11f

BRIGHT

ITERATION 8

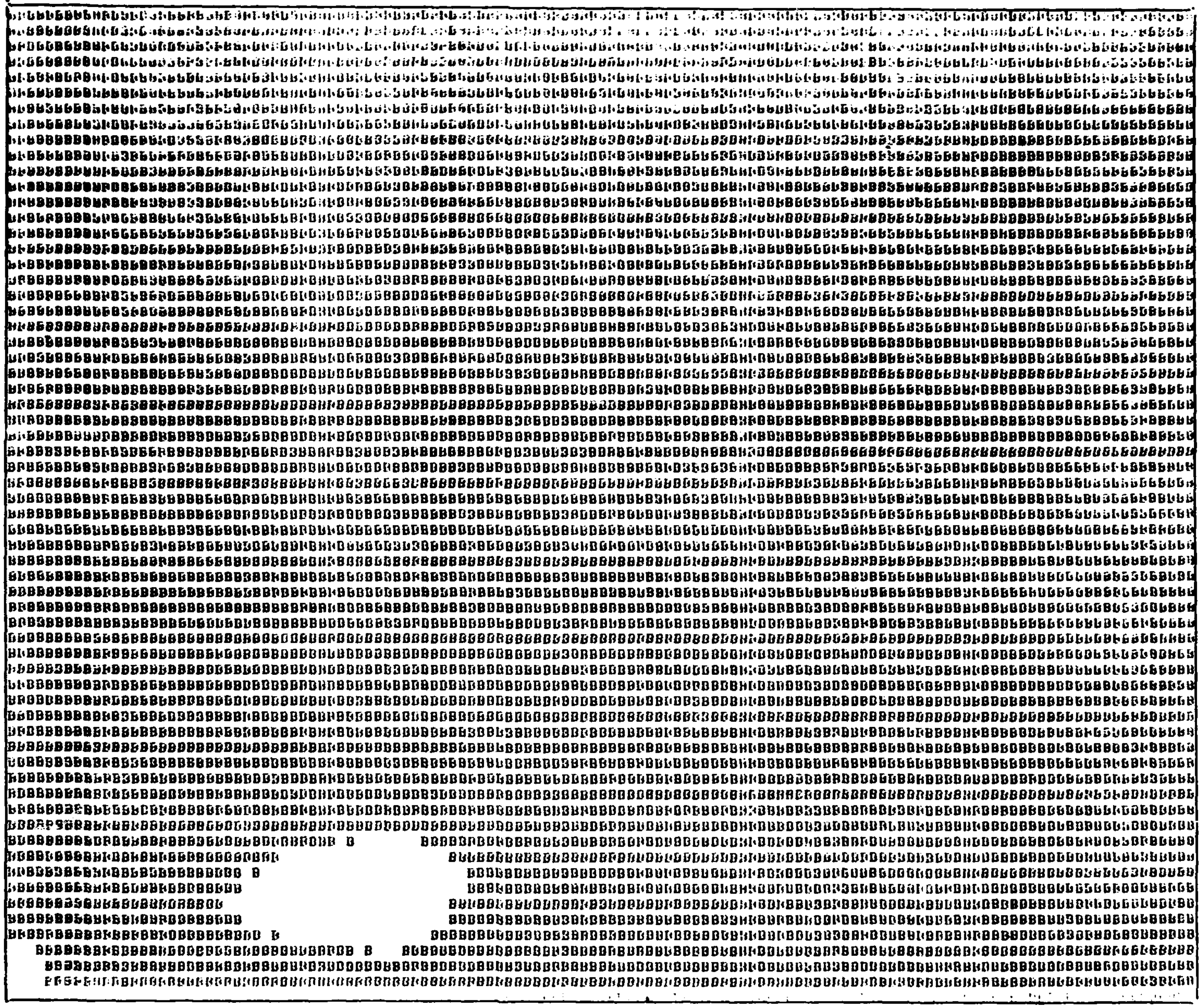


Figure 11g

CLUTTER

ITERATION 8

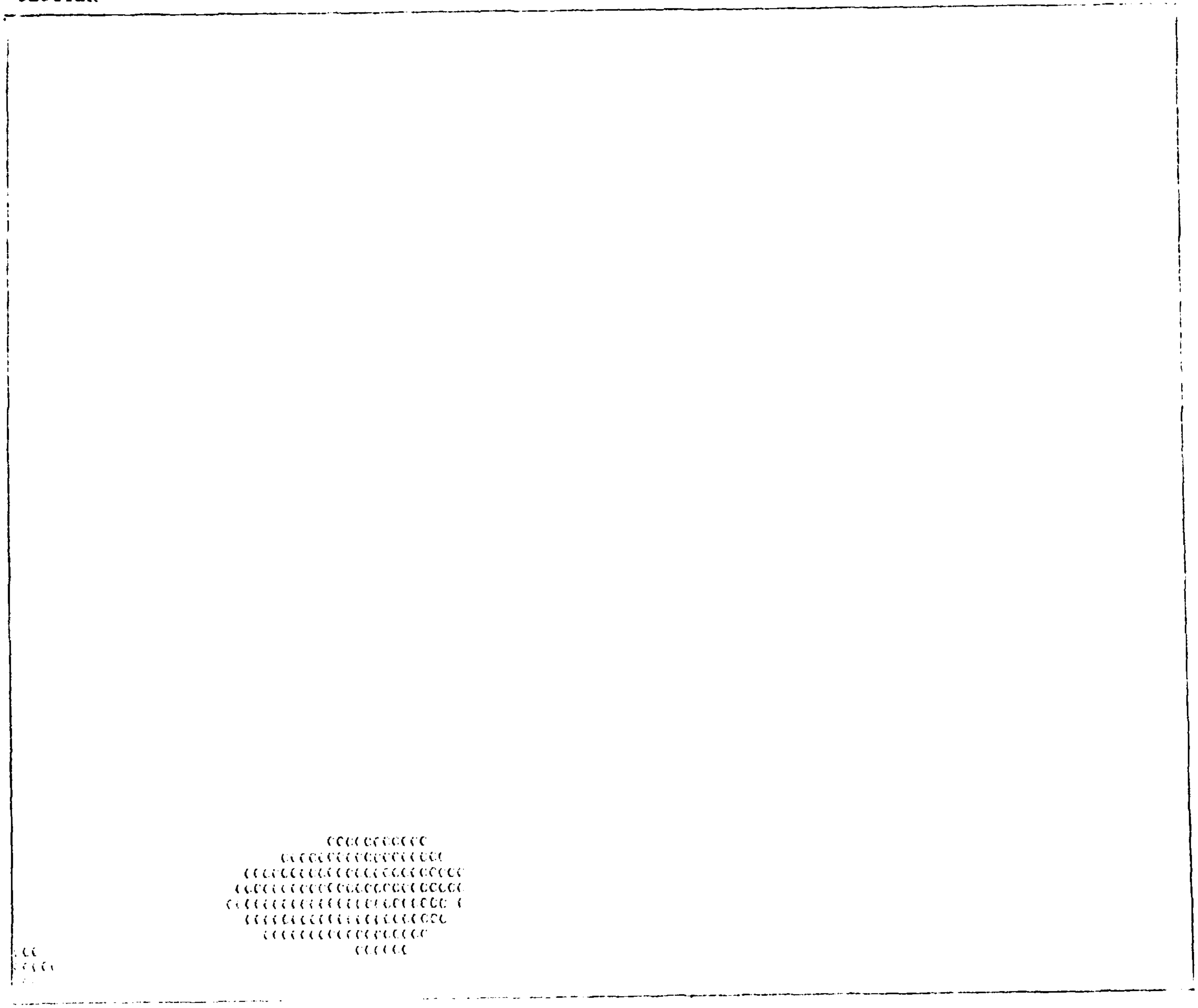


Figure 11h

BRIGHT

ITERATION 100

Figure 11i

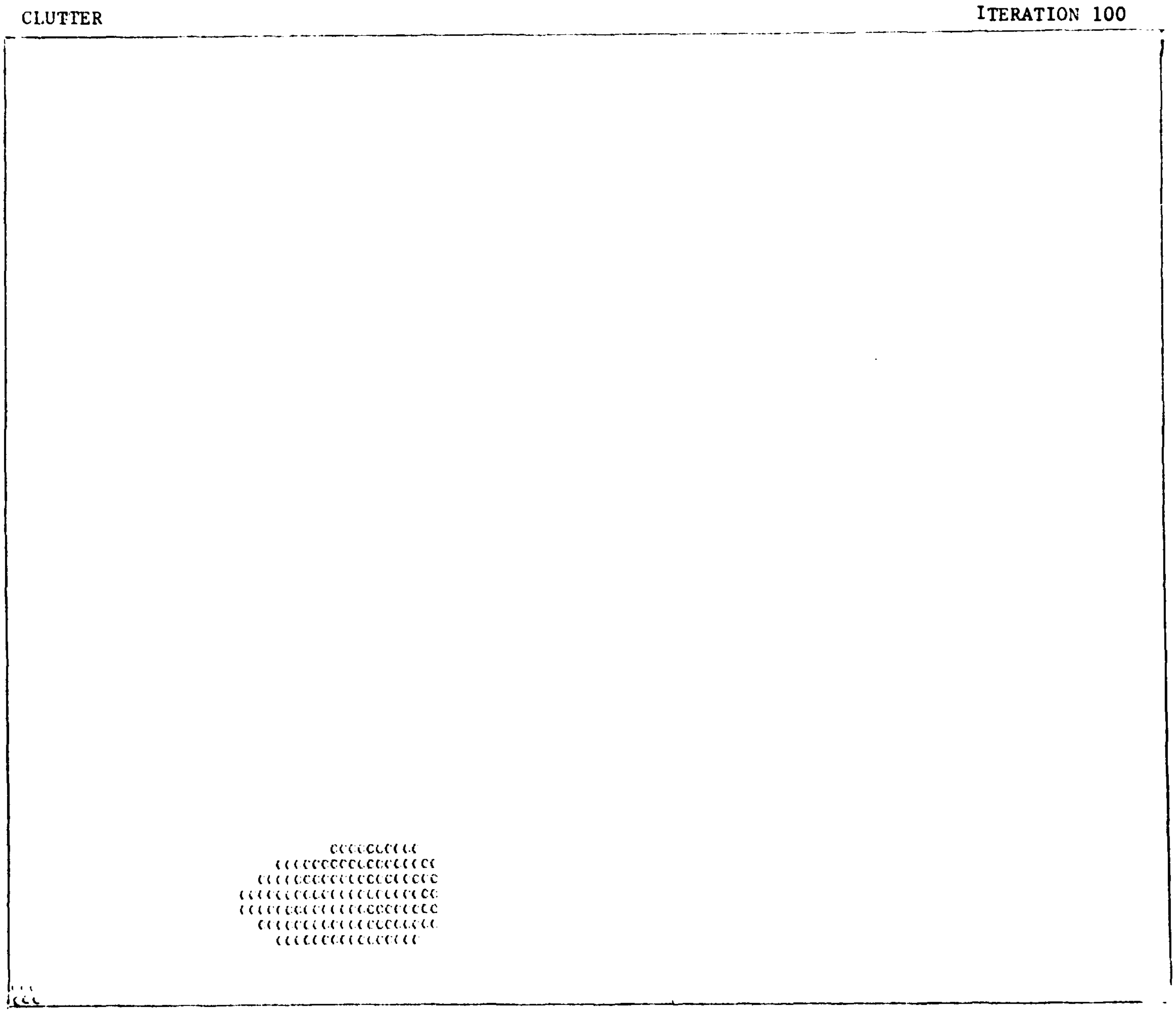


Figure 11j

```
FILE NAME= P26.DAT
BRIGHT/DARK/CLUTTER RELAXATION (64 BY 64)
KB= 1.00KBC= 0.00
KD= 1.00KCD= 0.00
KC= 1.00KBCD= 0.00
DELTA= 0.50
```

Figure 12

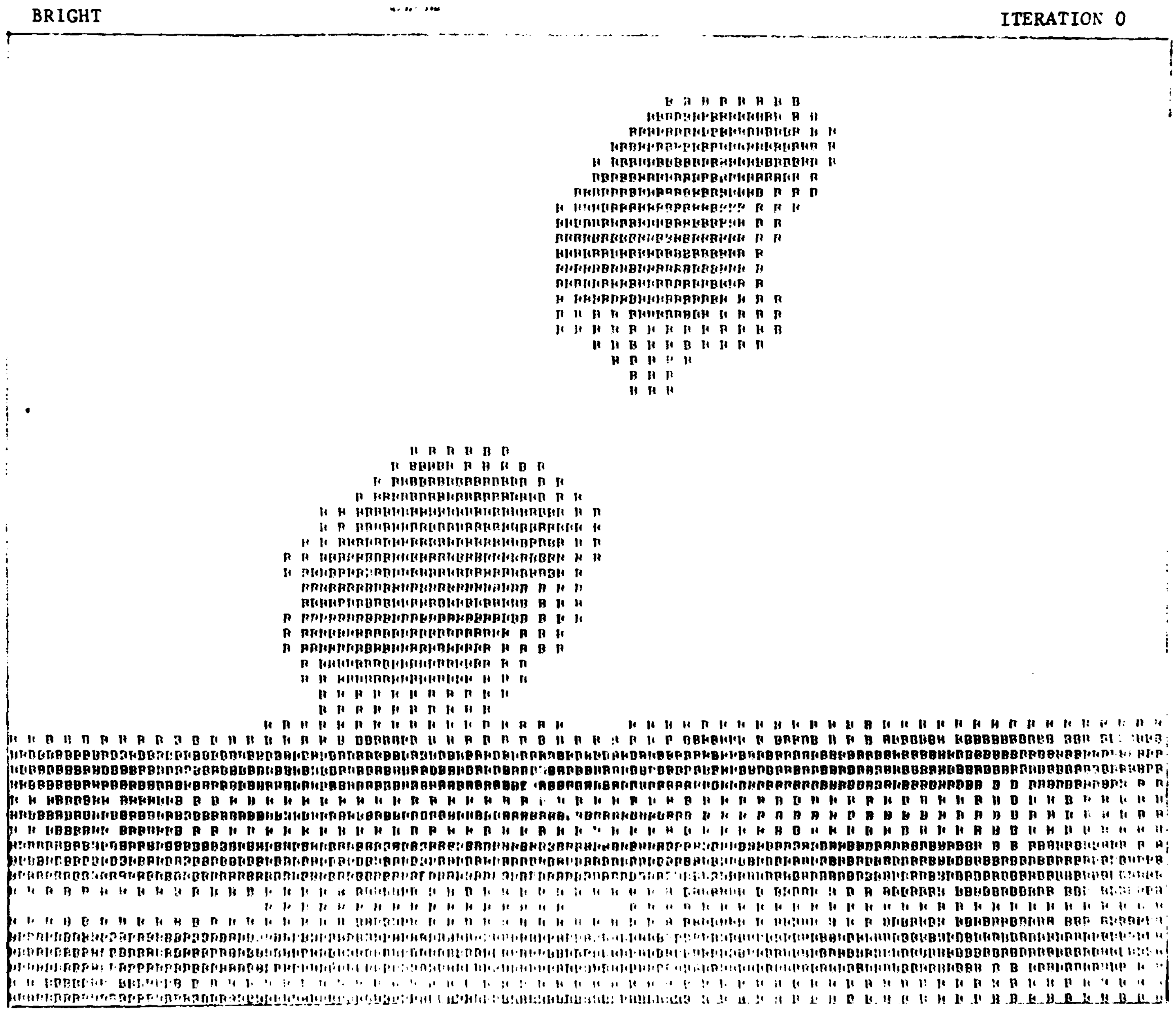


Figure 12a

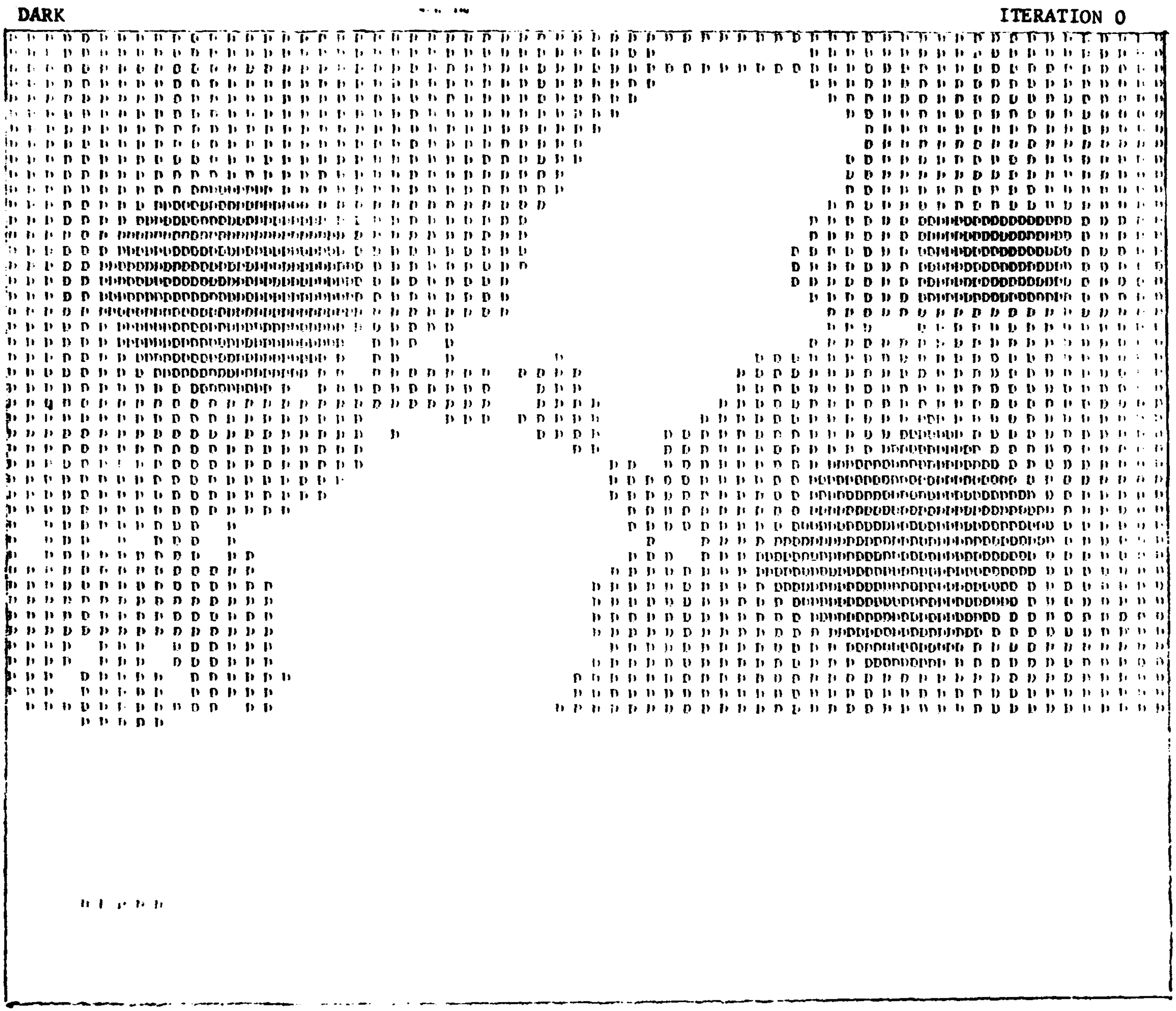


Figure 12b

CLUTTER ITERATION 0

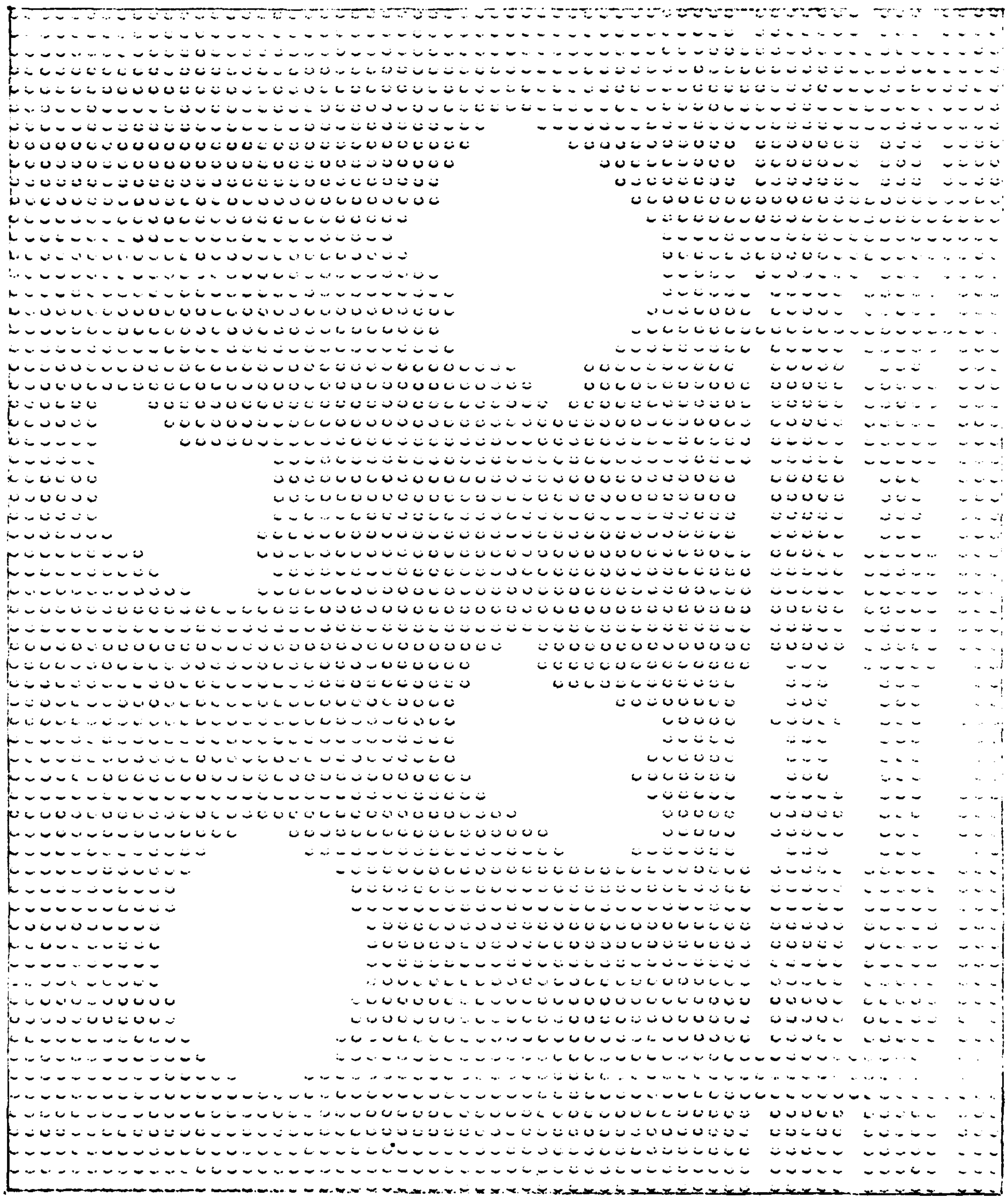


Figure 12c

BRIGHT

ITERATION 1

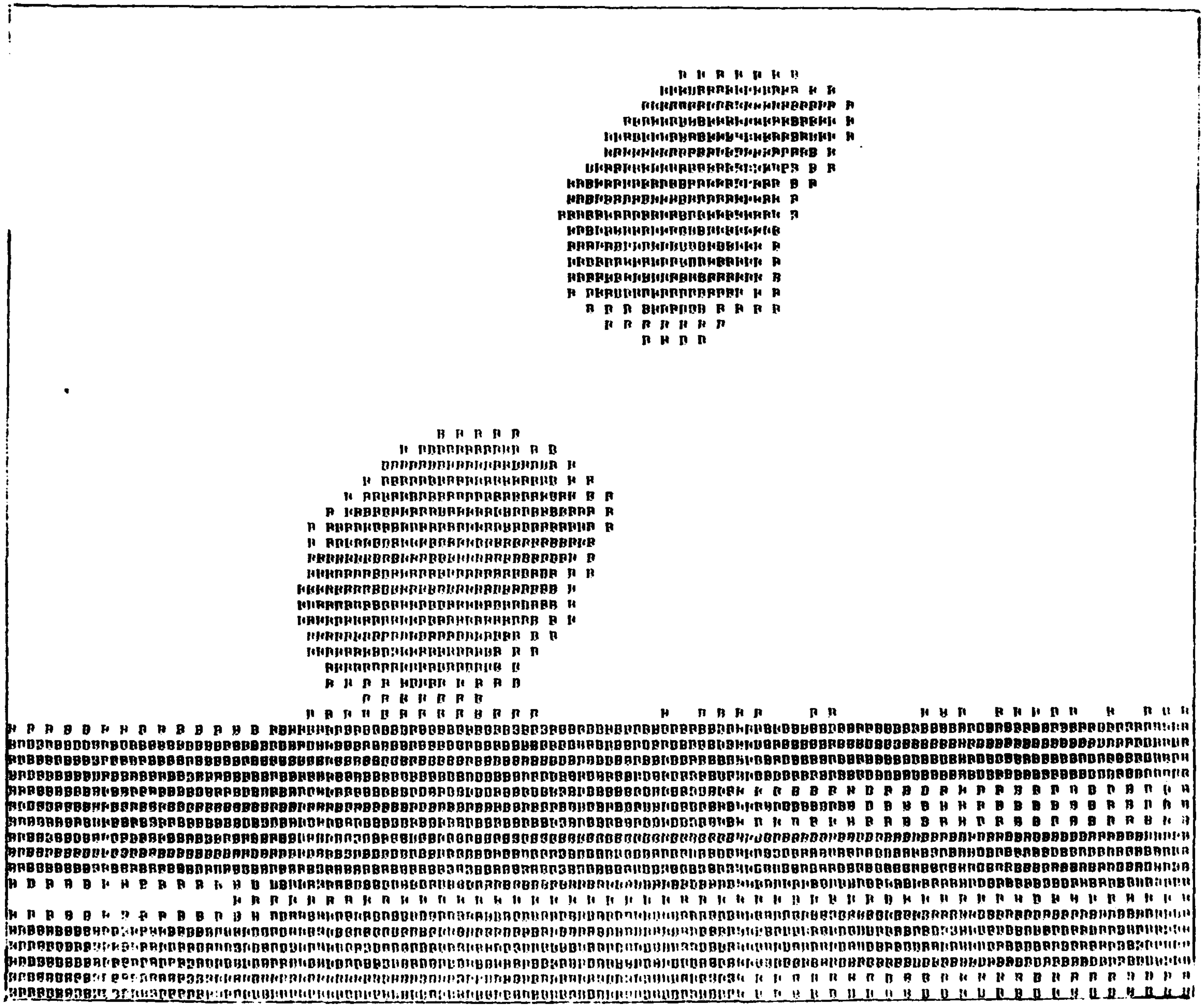


Figure 12d

DARK

ITERATION 1

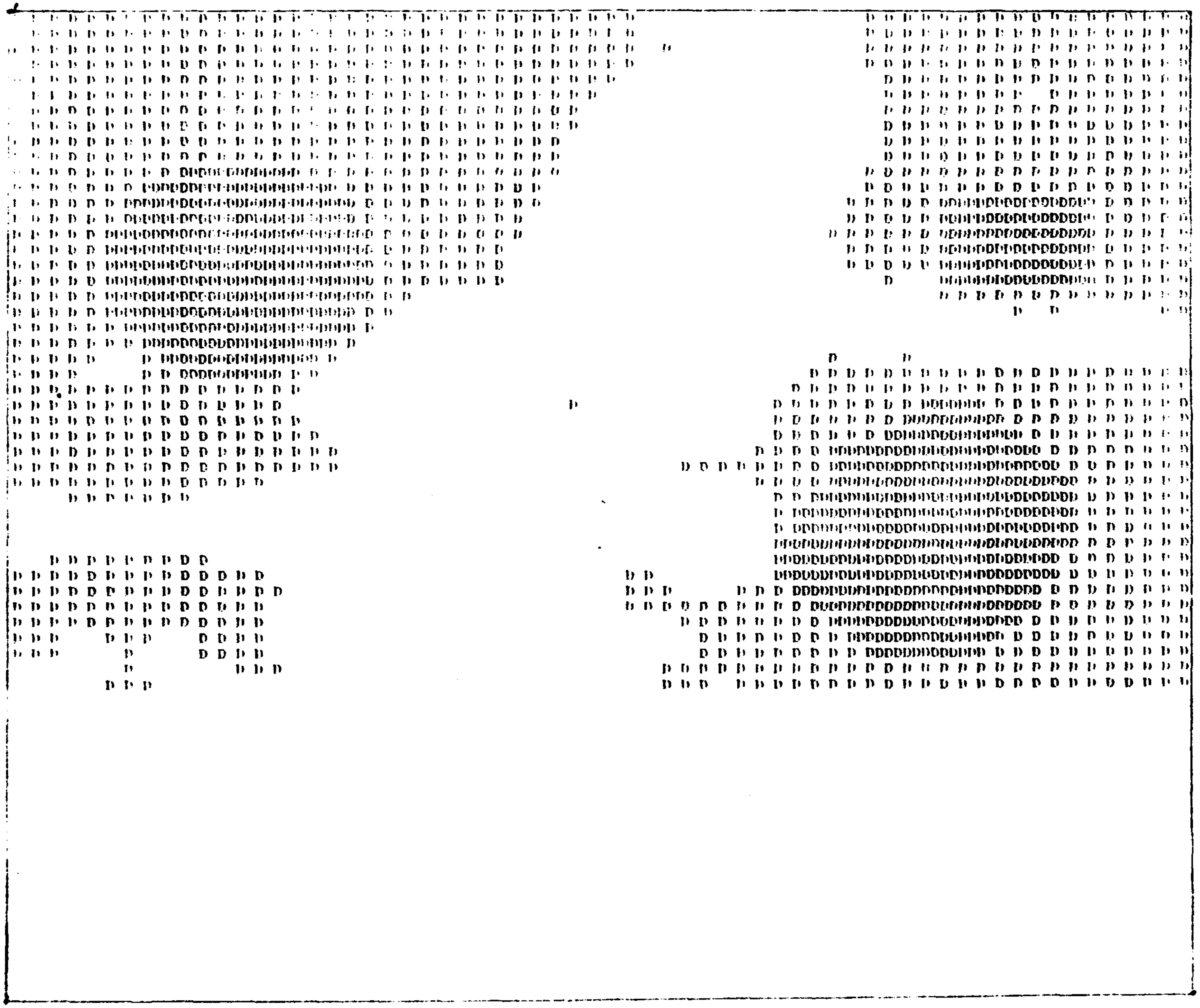


Figure 12e

CLUTTER ITERATION 1

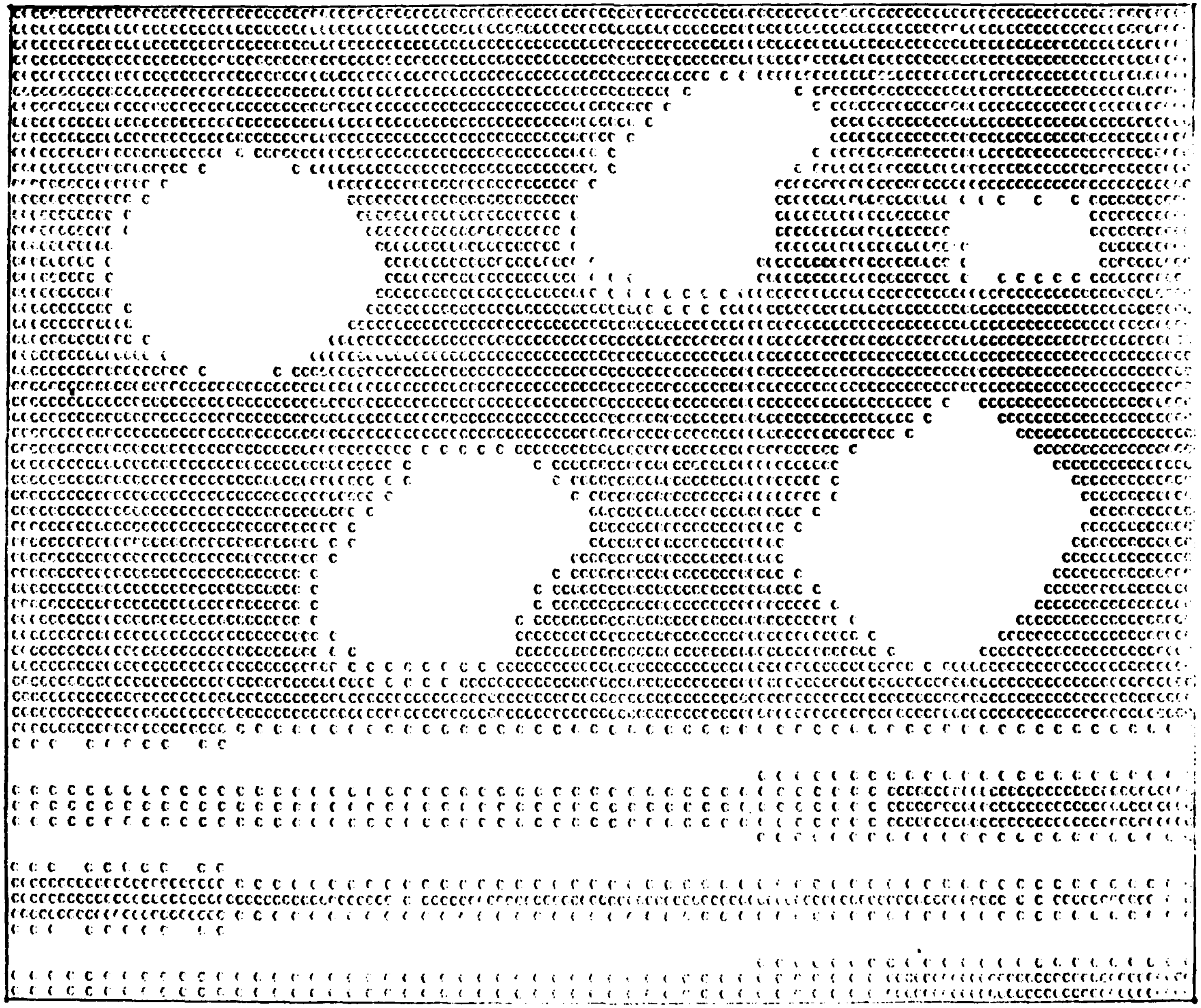


Figure 12f

BRIGHT

ITERATION 2

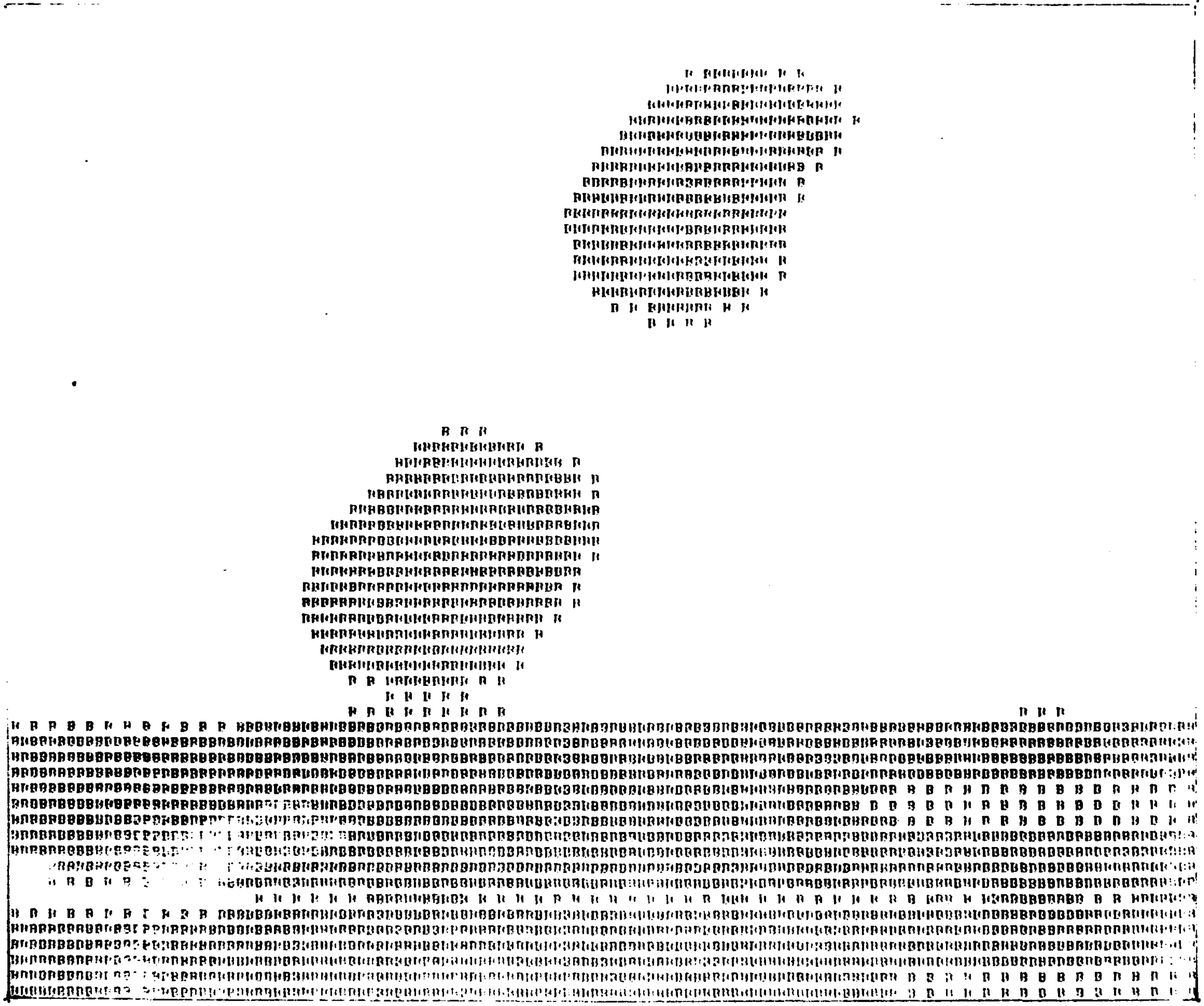


Figure 12g

DARK

ITERATION 2

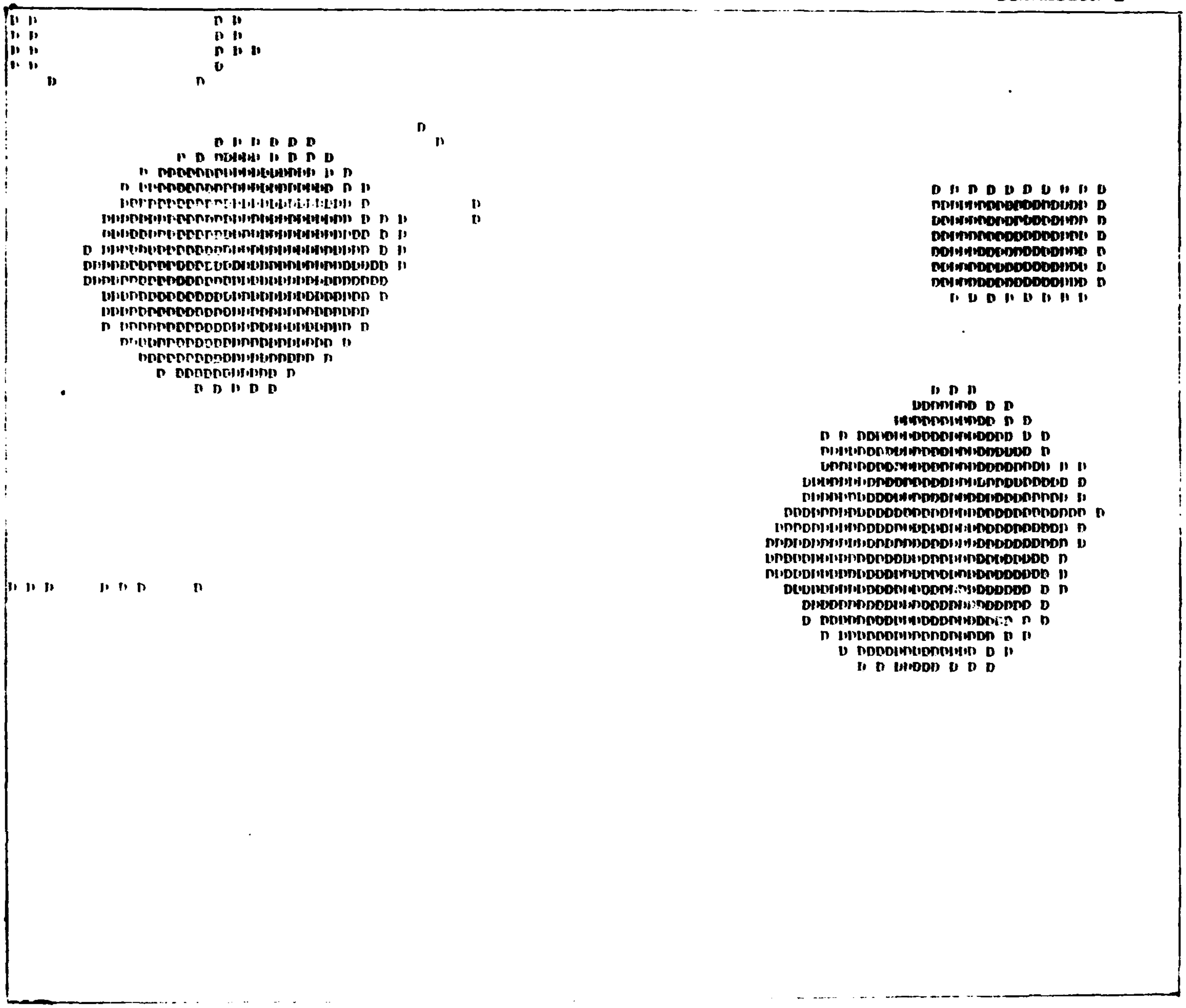


Figure 12h

CLUTTER ITERATION 2

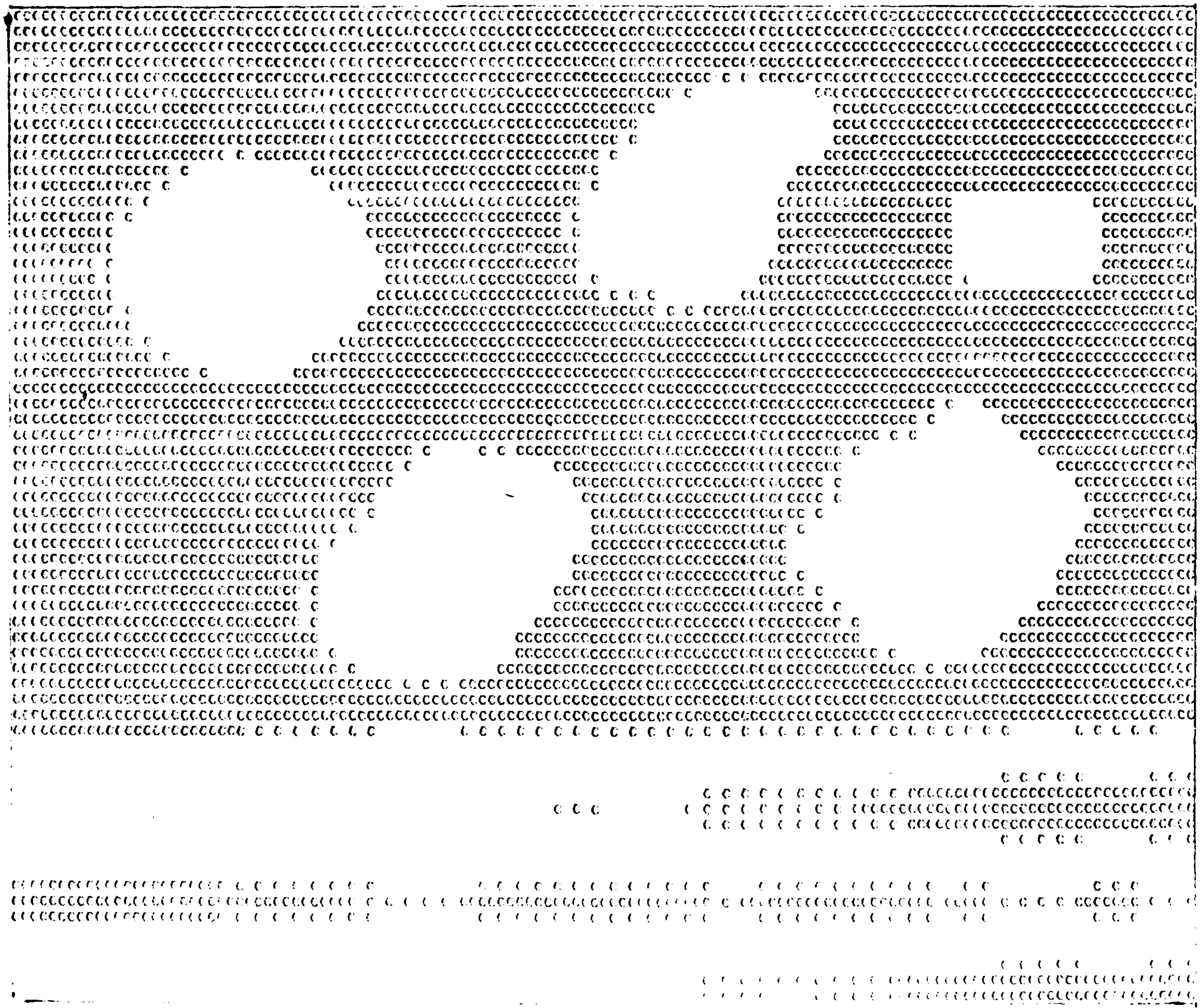


Figure 12i

BRIGHT

ITERATION 4

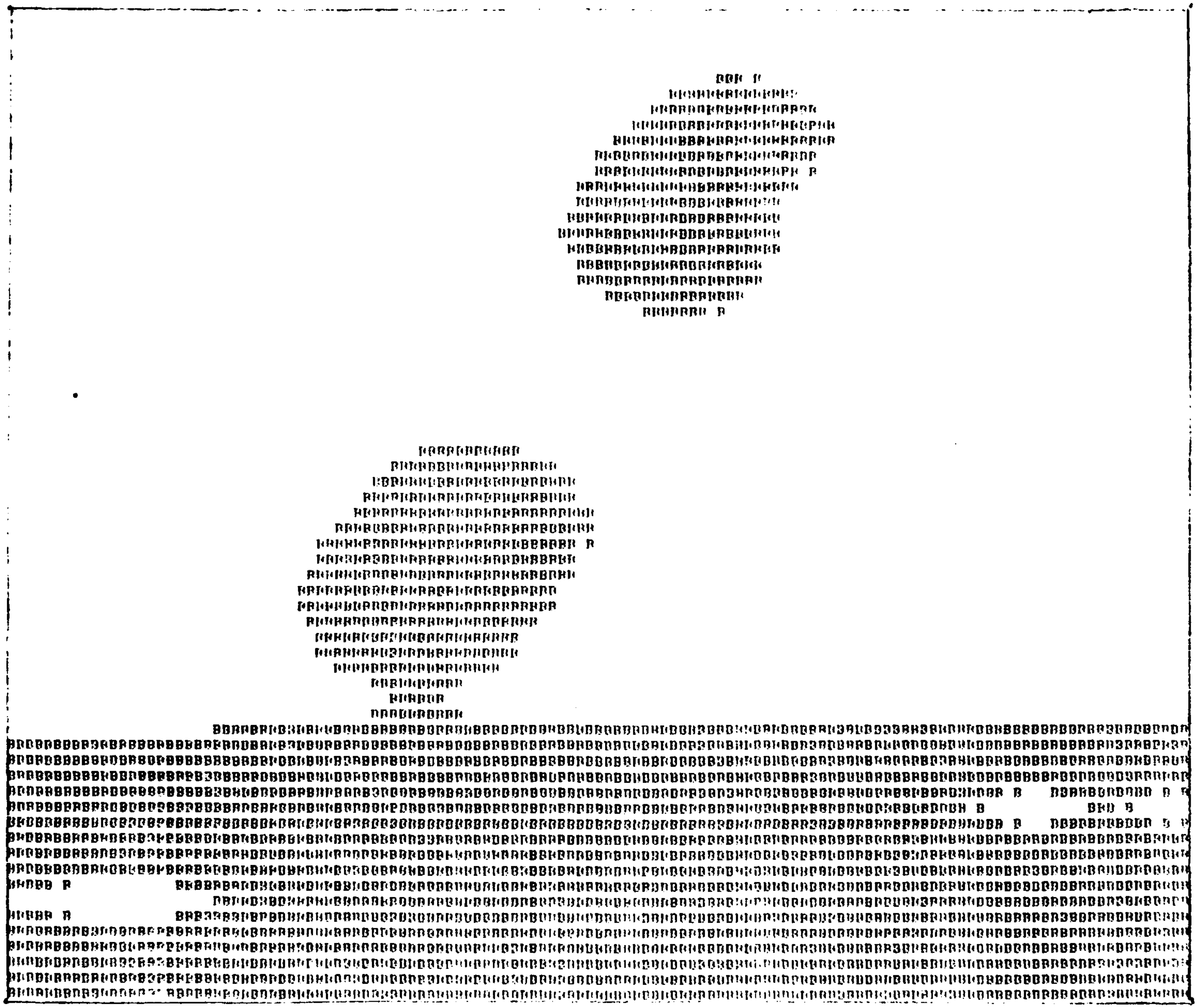


Figure 12j

4

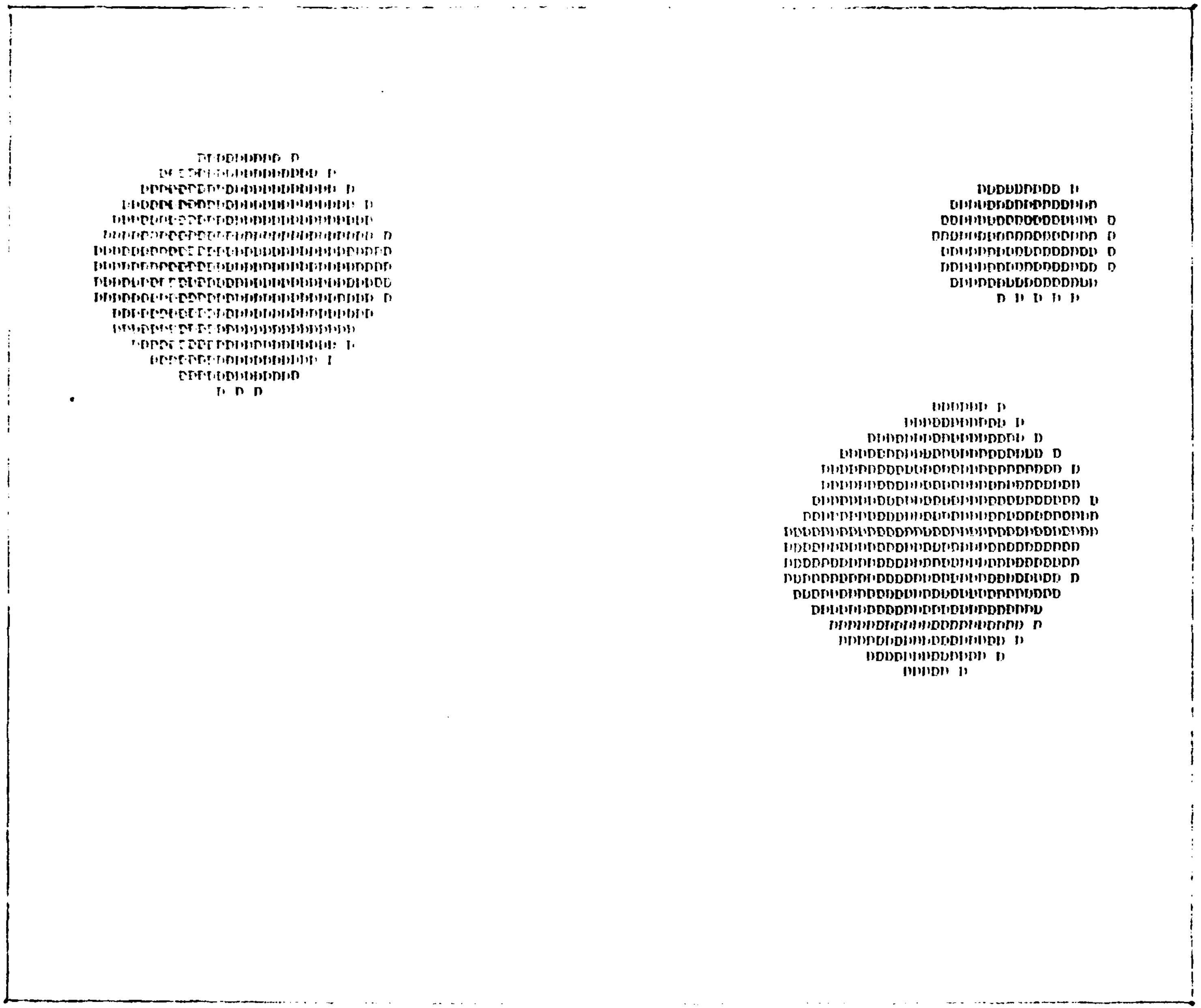


Figure 12k

CLUTTER

ITERATION 4

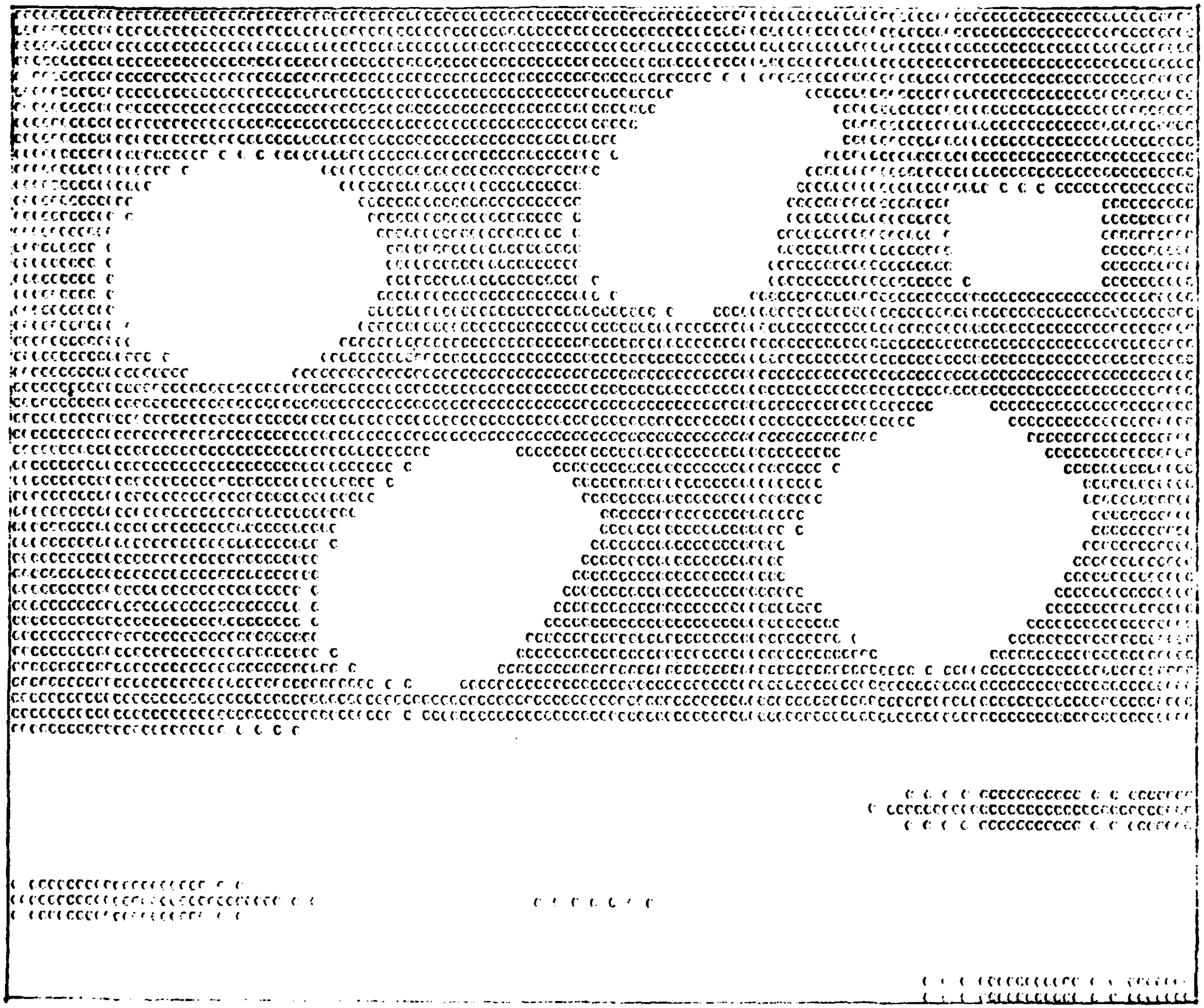


Figure 12-1

BRIGHT

ITERATION 8

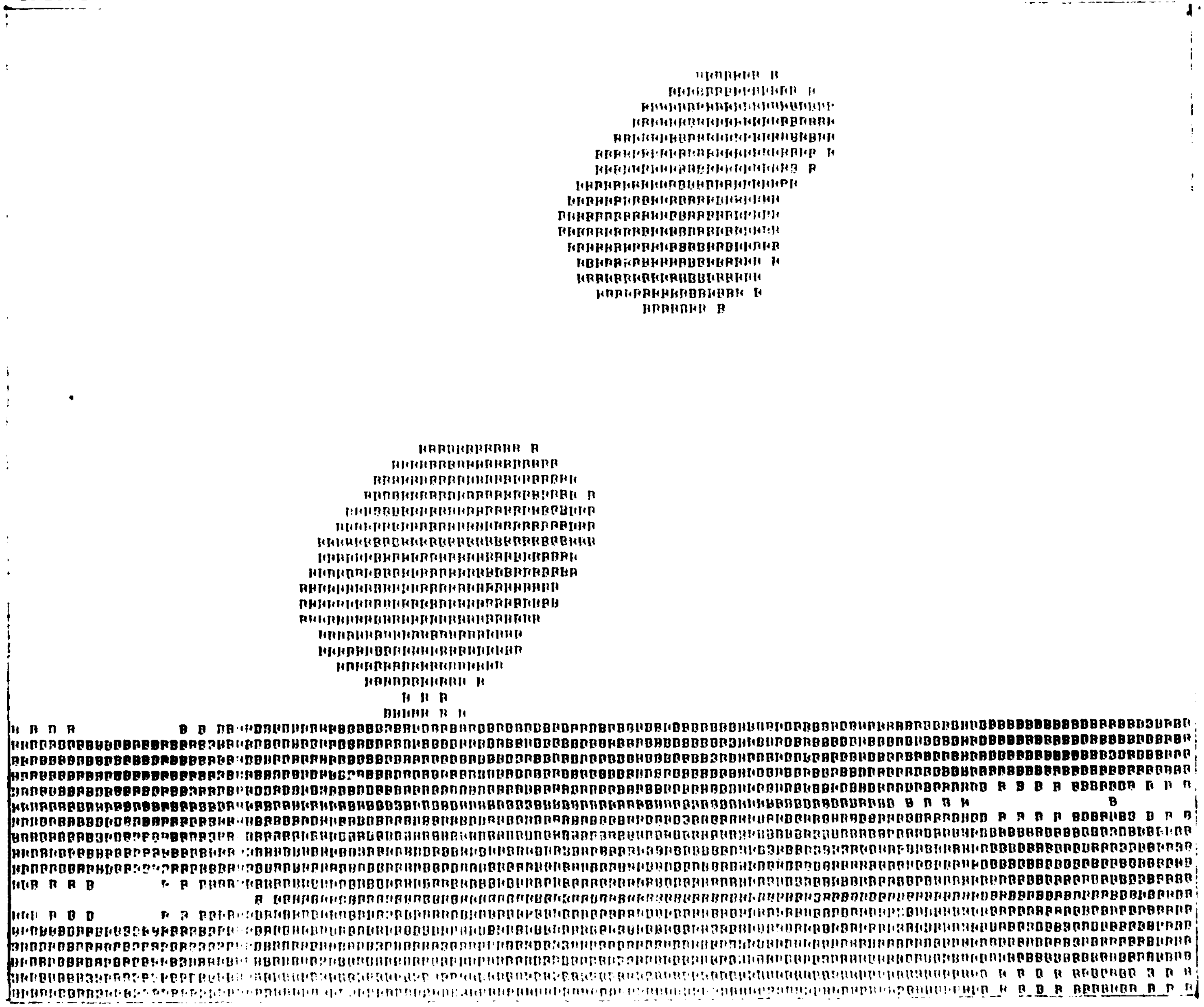


Figure 12m

DARK ITERATION 8

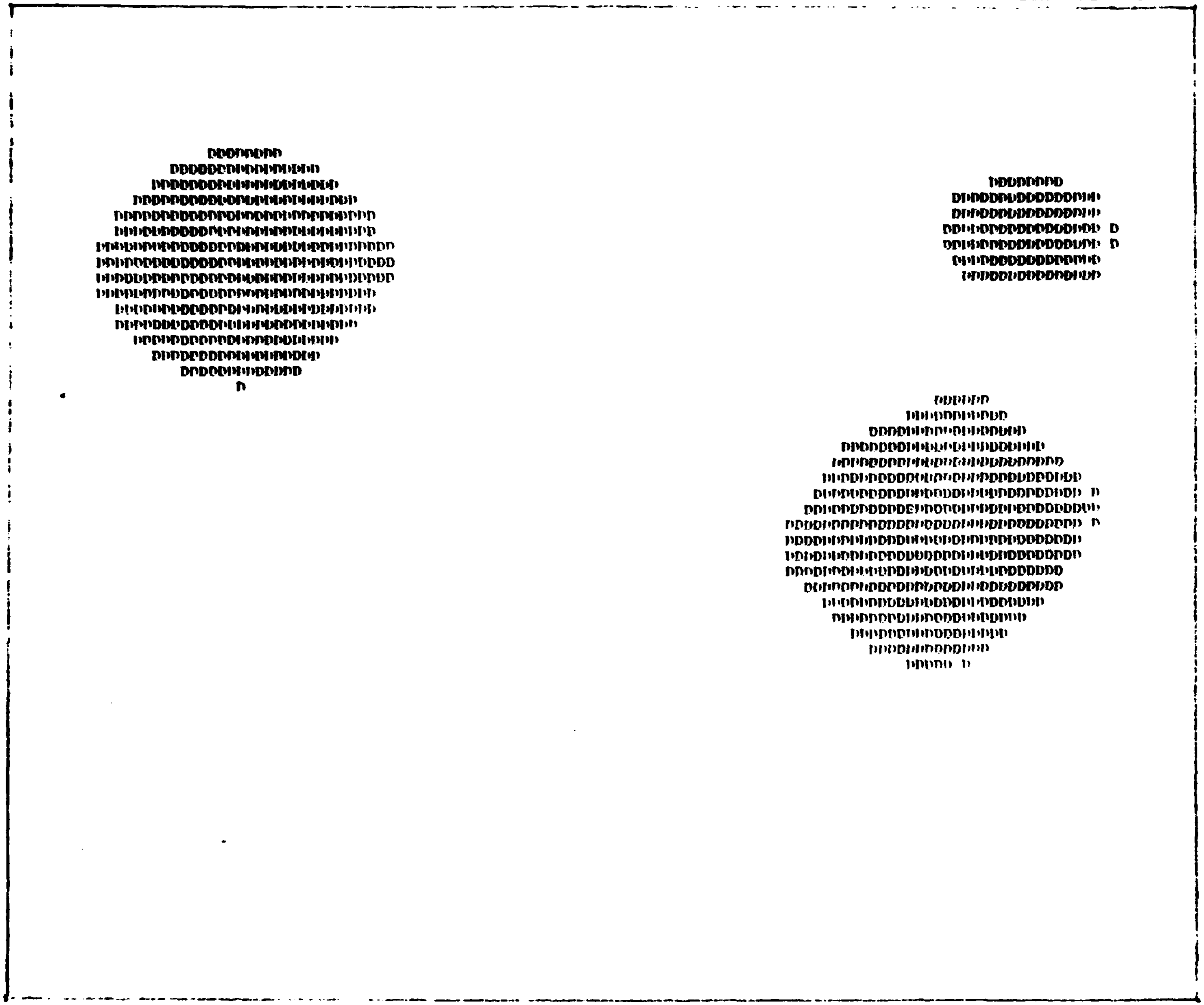


Figure 12n

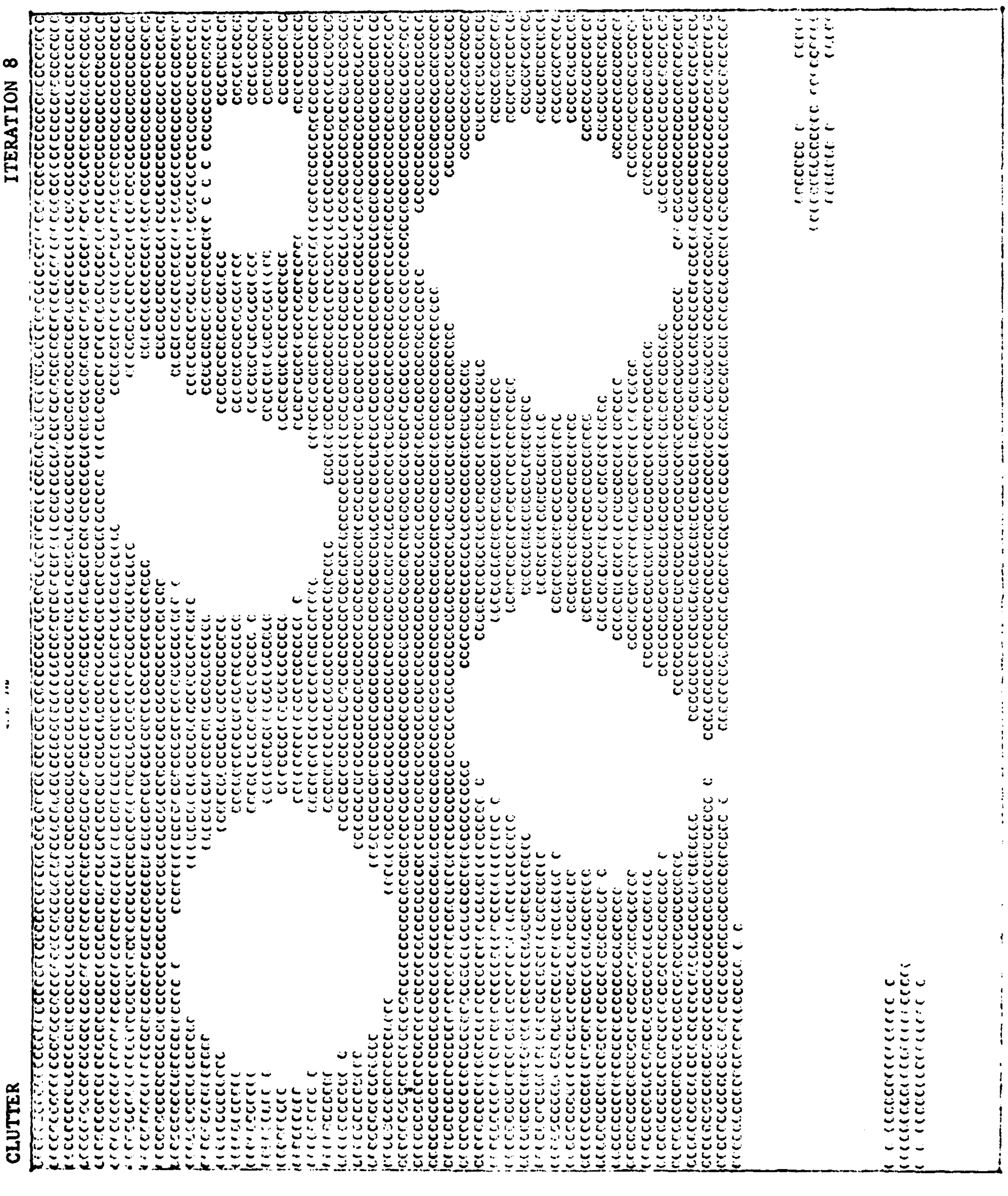


Figure 12o

BRIGHT

ITERATION 100

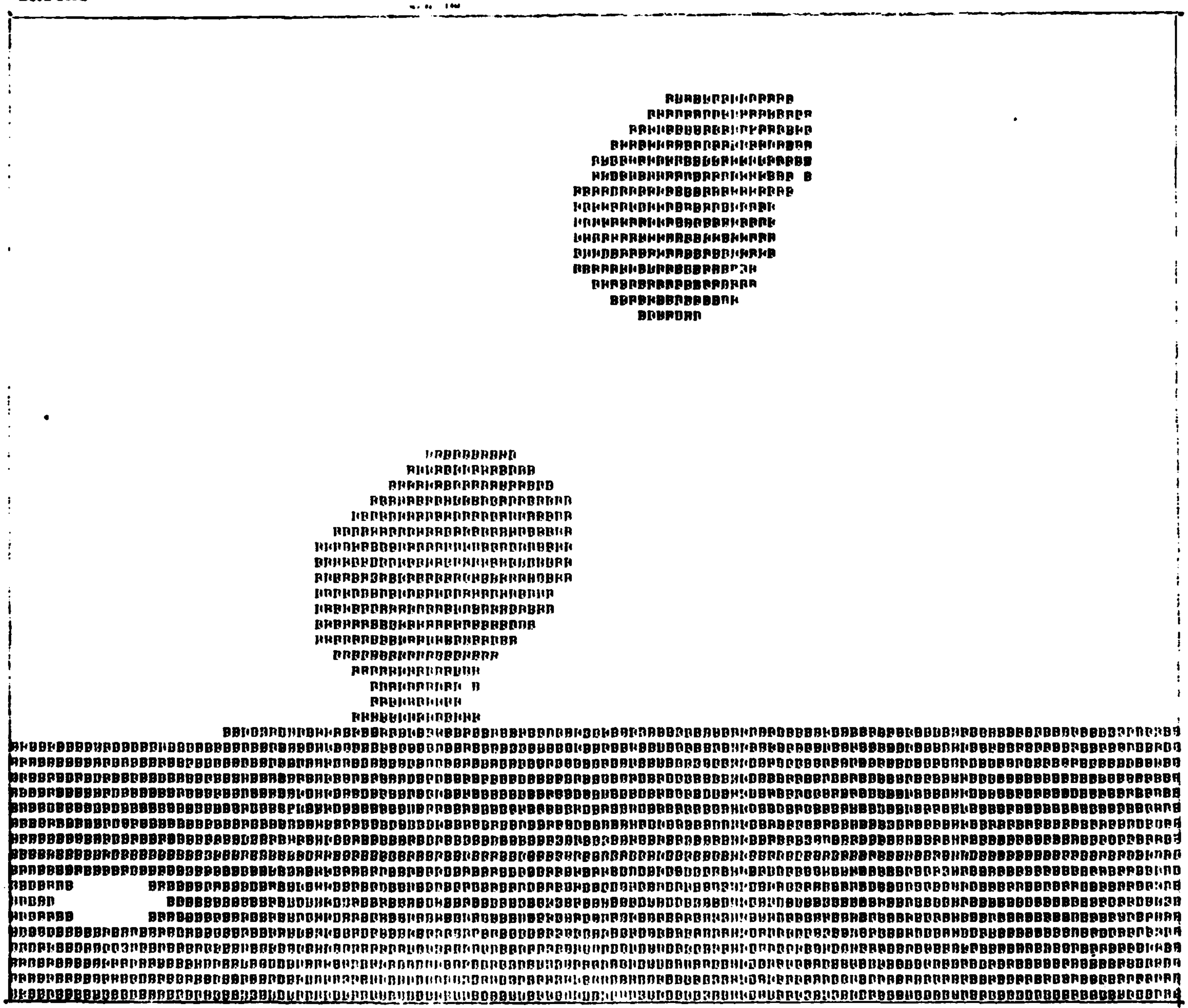


Figure 12p

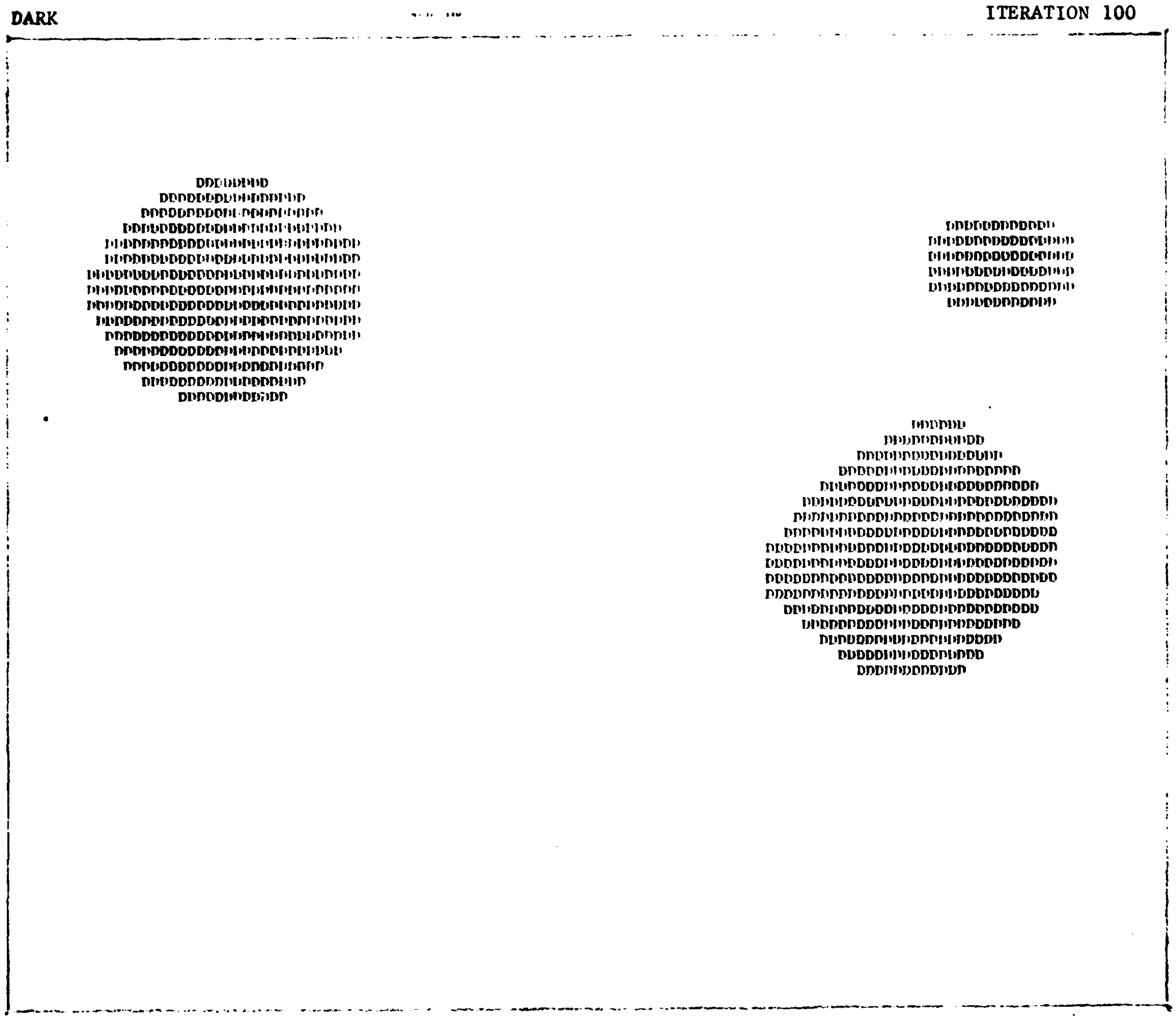


Figure 12q

CLUTTER

ITERATION 100

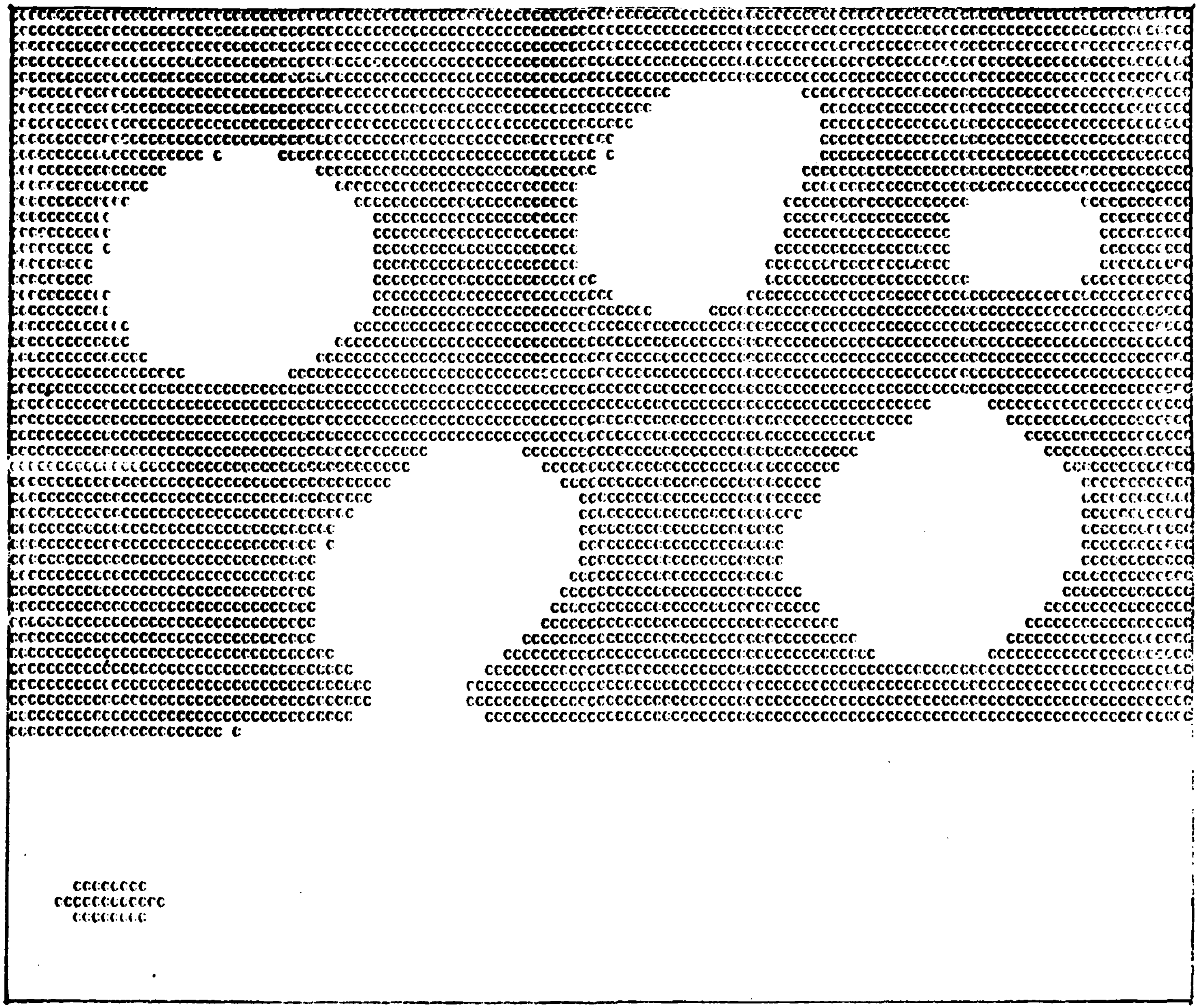


Figure 12r